

Light Field Imaging for Deflectometry

Zur Erlangung des akademischen Grades eines

DOKTORS DER INGENIEURWISSENSCHAFTEN (Dr.-Ing.)

von der KIT-Fakultät für
Elektrotechnik und Informationstechnik
des Karlsruher Instituts für Technologie (KIT)
angenommene

DISSERTATION

von

David Uhlig, M.Sc.

geb. in Marburg

Tag der mündlichen Prüfung:	18. November 2022
Hauptreferent:	Prof. Dr.-Ing. Michael Heizmann, KIT
Korreferent:	Prof. Dr.-Ing. Rainer Tutsch, TU Braunschweig



This document is licensed under a Creative Commons
Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0):
<https://creativecommons.org/licenses/by-sa/4.0/deed.en>

Preface

This thesis was written during my time at the Institute of Industrial Information Technology (IIIT) of the Karlsruhe Institute of Technology (KIT) and it would not have been possible without the help and support of numerous people. I would like to thank them all at this point.

I am grateful to Prof. Dr.-Ing. Michael Heizmann for providing me with the opportunity to work as a research assistant and for his steady support and direction as a supervisor, leading me to my present work. I also want to thank Prof. Dr.-Ing. Rainer Tutsch for taking over the co-review.

I would also like to thank all staff members of the institute for their support in bureaucratic matters and in the mechanical and electrical construction of the prototypes created during the work.

I am grateful to the many students whom I was allowed to supervise during my time at the institute. Valuable contributions and ideas for this work were generated in numerous technical discussions.

I would like to express my immense gratitude to all my research colleagues for the valuable moments we shared together, both within and outside the institute. Without their cooperative spirit and the productive exchange, this work would not have been possible. In particular, I would like to thank Johannes Anastasiadis, Matthias Bächle, Manuel Bihler, Daniel Diaz Ocampo, Muen Jin, Fabian Leven, Lanxiao Li, Theresa Panther, Max Schambach, Markus Schwabe, Erik Tabuchi Barczak, and Hannes Weinreuter for proofreading this thesis.

Finally, I am deeply thankful to my family and girlfriend for being my source of strength and providing me with unending support throughout my PhD journey.

Karlsruhe, February 2023

David Uhlig

Contents

Preface	i
Nomenclature	vii
1 Introduction	1
1.1 Contributions	4
1.2 Overview	6
2 Preliminaries	9
2.1 Operators	9
2.2 Rotation Parametrization	10
2.3 Line Parametrization and Plücker Coordinates	12
2.4 Surface Parametrization	14
2.5 Primal-Dual Optimization	15
3 Background	17
3.1 Deflectometry	17
3.1.1 Measurement Principle	18
3.1.2 Related Works	20
3.2 Light Fields	24
3.2.1 Light Field Acquisition	27
3.2.2 Related Works	30
4 Deflectometric Registration	33
4.1 Phase-Shift Coding	36
4.1.1 Phase Uncertainty	38
4.2 Principles of Phase Unwrapping	41
4.2.1 Temporal Phase Unwrapping	42
4.3 Improving the Phase Unwrapping Algorithms	46
4.3.1 Weighted Circular Mean	48
4.3.2 Modified Hierarchical Unwrapping	49

4.3.3	Modified PDM Unwrapping	50
4.4	Probabilistic Approach for Temporal Phase Unwrapping	51
4.4.1	Probability Density Function of Phase-Shift Coding	52
4.4.2	Compound Probability Density Function	55
4.4.3	Maximum-Likelihood Phase Unwrapping	58
4.4.4	Spatio-Temporal Phase Unwrapping	61
4.5	Evaluation	64
4.5.1	Qualitative Comparison	64
4.5.2	Robustness Against Noise	69
4.5.3	Comparison of Different Phase Noise Models . . .	74
4.5.4	Phase Map Reconstruction	76
4.6	Summary	81
5	System Calibration	83
5.1	Principles of Camera Calibration	84
5.2	Generic Camera Calibration	86
5.2.1	The Generic Camera Model	87
5.2.2	Related Works	88
5.2.3	Alternating Minimization-Based Calibration . . .	89
5.2.4	Generic Ray Estimation	93
5.2.5	Generic Pose Estimation	96
5.2.6	Convergence, Acceleration and Initialization . . .	100
5.2.7	Normalizing the Ray Bundle	102
5.3	Calibration of the Reference Target	104
5.3.1	Reference Surface Model	106
5.3.2	Parameter Estimation	109
5.4	Calibration of the Deflectometry Setup	111
5.4.1	Linear Solution	112
5.4.2	Nonlinear Optimization	115
5.5	Evaluation	116
5.5.1	Error Metrics	119
5.5.2	Initialization of the Alternating Minimization . .	119
5.5.3	Convergence of the Alternating Minimization . .	121
5.5.4	Required Number of Poses	123
5.5.5	Evaluation of the Calibration Error	124
5.5.6	Mirror-Based Pose Estimation	131

5.5.7	Shape Estimation of the Reference Target	133
5.6	Summary	134
6	Light Field Reconstruction	135
6.1	Related Works	137
6.2	Generic Light Field Reconstruction	139
6.2.1	Parametrization of Light Field Coordinates	140
6.2.2	Weighted Interpolation of Irregular Data	145
6.2.3	Intrinsic Camera Parameters	147
6.3	Evaluation	151
6.3.1	Light Field Reconstruction: Lytro Illum	152
6.3.2	Comparison of Angular Sampling Grids	164
6.3.3	Super-Resolution through Implicit Ray Interpolation	165
6.3.4	Light Field Reconstruction: Raytrix R5	167
6.3.5	Light Field Reconstruction: K Lens	171
6.3.6	Camera Intrinsic and Calibration Error	173
6.4	Summary	179
7	Specular Surface Reconstruction	181
7.1	Deflectometric Regularization	182
7.1.1	Light Field Depth-Based Regularization	183
7.1.2	Light Field Multi-View-Based Regularization	187
7.2	Surface Reconstruction	191
7.2.1	Surface Reconstruction by Depth and Normal Fusion	193
7.2.2	Multi-Depth Reconstruction	197
7.2.3	Multi-View Reconstruction	198
7.3	Evaluation	199
7.3.1	Regularization	200
7.3.2	Multi-Depth Reconstruction	206
7.3.3	Multi-View Reconstruction	209
7.3.4	Influence of the Calibration	213
7.4	Summary	216
8	Conclusion	217
8.1	Summary	217

8.2	Outlook	220
9	Appendix	223
9.1	Calibration	223
9.1.1	Variables	223
9.1.2	Riemannian Gradient and Hessian on $SO(3)$. . .	224
9.1.3	Proofs	226
	Bibliography	229
	List of Publications	254
	List of Supervised Theses	255

Nomenclature

Common abbreviations

Abbreviation	Description
<i>cf.</i>	<i>Confer</i> (lat. refer to)
<i>e.g.</i>	<i>Exempli gratia</i> (lat. for example)
<i>et al.</i>	<i>Et alii</i> (lat. and others)
<i>etc.</i>	<i>Et cetera</i> (lat. and so forth)
<i>i.e.</i>	<i>Id est</i> (lat. that means)
AM	Alternating minimization
CNN	Convolutional neural network
EPI	Epipolar plane image
GT	Ground truth
JSD	Jensen-Shannon distance
LF	Light field
MAE	Mean absolute error
MLA	Microlens array
MSE	Mean-squared error
PDM	Projection distance minimization
PnP	Perspective-n-Point
PV	Peak-to-valley distance
RMSE	Root-mean-squared error
SAI	Subaperture image
SNR	Signal-to-noise ratio
TGV	Total Generalized Variation
TV	Total Variation

Symbols

Latin letters

Symbol	Description
a	Scalar
\mathbf{a}	Vector
$\hat{\mathbf{a}}$	Unit vector
\mathbf{A}	Matrix
\mathbf{d}	Ray direction
f_s, f_t, f	Focal length
f	Frequency
\mathbf{g}	Surface gradient
\mathbf{g}_m	Deflectometrically measured surface gradient
\mathbf{I}	Identity matrix
\mathbf{K}	Intrinsic camera matrix
L	Light field
\mathbf{l}	Plücker line
\mathcal{L}	Set of rays
\mathbf{m}	Ray momentum
$\mathcal{N}_{s,t,u,v}^m$	Set of nearest neighbors
\mathbf{n}	Surface normal
\mathbf{n}_m	Deflectometrically measured surface normal
n	Refraction index
\mathbf{p}	Monitor features in the camera coordinate system
\mathbf{s}	Surface point
\mathcal{R}	Set of rotation matrices
\mathbf{R}	Rotation matrix
$\tilde{\mathbf{R}}$	Reflected rotation matrix
\mathbf{r}	Vectorized rotation matrix
r, ϕ	Angular coordinates in polar parametrization
N_r, N_ϕ	Angular resolution of polar parametrization
s, t	Pixel coordinates or spatial light field coordinates
N_s, N_t	Spatial light field resolution
s_x	Unwrapping success rate
\mathbf{t}	Translation vector

Symbol	Description
$\tilde{\mathbf{t}}$	Reflected translation vector
\mathcal{T}	Set of translation vectors
\mathbf{u}	2D Pixel coordinate as vector
u, v	Angular light field coordinates
N_u, N_v	Angular light field resolution
\mathbf{x}, x, y	Monitor feature coordinates
$\mathbf{y}_0, \mathbf{y}_1$	Dual variables
z	Depth

Greek letters

Symbol	Description
ε	Point-to-ray distance
ϵ_x	Unwrapping error
λ	Wavelength
σ_I	Standard deviation of image noise
σ_φ	Phase uncertainty
σ_{ik}	Coordinate uncertainty
φ	Wrapped phase
Φ	Unwrapped phase
Ψ_k	Phase offset

Superscripts

Index	Description
$(\bullet)^T$	Transposed
$(\bullet)^*$	Adjoint
$(\bullet)^{(n)}$	Index of iteration

Subscripts

Index	Description
$(\bullet)_i$	Pixel index
$(\bullet)_k$	Pose index
$(\bullet)_{uv}$	Subaperture image index
$(\bullet)_{\text{direct}}$	Direct depth regularization
$(\bullet)_{\text{indirect}}$	Indirect depth regularization
$(\bullet)_{\text{ML}}$	Maximum likelihood
$(\bullet)_{\text{multi}}$	Multi-view regularization
$(\bullet)_{\text{PM}}$	Phase noise model
$(\bullet)_{\text{reflect}}$	Depth of reflection
$(\bullet)_{\text{vM}}$	<i>von Mises</i>
$(\bullet)_{\text{WN}}$	Wrapped normal

Mathematical operators

Operator	Description
$\hat{\cdot}$	Vector normalization
$\langle \cdot, \cdot \rangle$	Scalar product
$\ \cdot\ $	Euclidian norm
$\ \cdot\ _0$	l_0 -norm
$\ \cdot\ _\infty$	l_∞ -norm
\otimes	Kronecker product
$\text{vec}(\cdot)$	Vectorization operator
$\text{mat}(\cdot)$	Inverse of vectorization operator
$[\cdot]_\times$	Skew-operator
∂_x	Partial derivative with respect to x
∇	Gradient operator
\mathcal{E}	Symmetrized gradient operator
div_∇	Divergence operator
$\text{div}_\mathcal{E}$	Divergence operator
D_ξ	Directional derivative in direction ξ

Operator	Description
$\text{grad}(\cdot)$	Riemannian gradient
$\text{Hess}(\cdot)$	Riemannian Hessian
$\text{Exp}(\cdot)$	Exponential map from tangent space to manifold
$\exp(\cdot)$	Exponential function
$\text{Log}(\cdot)$	Logarithmic map from manifold to tangent space
$\log(\cdot)$	Logarithm to base 10
$\ln(\cdot)$	Natural logarithm
$\text{prox}_f(\cdot)$	Proximal operator for function $f(\cdot)$
$\lceil \cdot \rceil$	Ceiling function
$\lfloor \cdot \rfloor$	Floor function
$\text{round}(\cdot)$	Round to closest integer
$\text{gcd}(\cdot, \cdot)$	Greatest common divisor
$d(\cdot, \cdot)$	Distance
$d^\circ(\cdot, \cdot)$	Circular distance
$\mathcal{M}^\circ(\cdot, \cdot)$	Weighted circular mean
$p(\cdot)$	Probability density function
$\mathcal{N}(\cdot)$	Density of normal distribution

1 Introduction

The increasing demand for specular freeform surfaces presented a major challenge for production and metrology in recent years. When inspecting specular reflective components, such as solar collectors, lens systems, wafers, or even telescope mirrors, the interest generally lies in manufacturing them as precisely as possible, which means that the exact geometric object dimensions have to be known. Reflection properties also play a crucial role for surfaces from the automotive industry, such as lacquered car body parts, or objects from the entertainment industry, since defects and flaws in the surface affect the aesthetics of the product to a great extent. The inspection of such surfaces is very demanding in practice. During the visual inspection of specularly reflective objects, in contrast to diffuse reflection, an observer does not see the surface itself, but the distorted mirror image of the environment. The reflective surface is virtually invisible to the observer. Automatic visual inspection, especially 3D measurement, therefore poses a great metrological challenge.

Deflectometric measurement methods use the law of reflection and knowledge of the arrangement between a camera and a pattern generator, *e.g.*, a liquid crystal display (LCD) monitor, to draw conclusions about the shape of the surface by means of observing the deformations of the mirror images. For automatic visual inspection and accurate 3D reconstruction, precise knowledge of the system parameters is required, *e.g.*, the size and position of the LCD monitor relative to the camera sensor, as well as the intrinsic camera parameters. If the 3D coordinates of a reference object point and its mirror reflection are known, the reflection point on the specular surface can in principle be calculated from them. However, the photographic measurement of an object point lacks distance information, that is, only directional information is available. Therefore, even with complete knowledge of the system, a single camera is generally not sufficient to calculate a unique surface from the measurement data. As a result, the solutions for the surface lie on a one-parametric solution

manifold and potential surface normals representing these surfaces can be calculated for any point in space. To find the true surface from this infinite variety, the problem must be regularized, where in principle, it would be sufficient to know only a single point. Starting from this point, the complete surface can then be reconstructed by integrating the deflectometrically measured normal field.

Obtaining an accurate reconstruction necessitates a precise measurement. Hence, sophisticated and highly specialized optical imaging devices are becoming increasingly important for high-precision manufacturing and environment perception. In particular, light field cameras are experiencing an ever-increasing interest in research and industry as they provide a four-dimensional light field of the scene instead of a two-dimensional image. The information captured by a light field camera in a single photographic exposure can be used to, for example, digitally refocus the image, extract depth information, or subsequently change the perspective of the scene. Light field cameras can therefore be regarded as compact 3D cameras. In contrast to camera arrays, in which the individual synchronized cameras each sample a part of the light field, the hardware requirements for light field cameras are significantly reduced, and they are far more robust against external influences. Even in a compact handheld camera design, they can capture several hundred perspectives of the scene in a single shot.

The advantages of light field cameras should therefore also be made accessible to the field of optical metrology. In particular, this thesis aims at combining light field imaging with deflectometry as this enables a variety of new measurement methods. The additional information compared to conventional cameras is to be used to regularize the ambiguity of the deflectometric measurement while providing a robust reconstruction of the surface. While light field cameras offer several advantages for deflectometry, they also introduce new difficulties and challenges. Thanks to their design, light field cameras have a very high depth of field, which improves the lateral resolution of the deflectometric measurement, but at the same time, the amount of captured light is reduced, resulting in higher noise sensitivity. In addition, the calibration of these cameras is, unfortunately, very difficult due to their complex structure and sophisticated optical design. To achieve the most accurate description of

the imaging process, advanced camera models and elaborate calibration techniques are required. To make light field cameras available for effective use in deflectometry, special focus is placed on four aspects in this work:

Registration: In deflectometry, the measurement of reference coordinates and thus the registration between camera pixels and points in the plane of the reference monitor provides the local slope of the surface. The angular resolution of the angle between the surface point and the reference point thereby determines the accuracy of the reconstruction. Therefore, for high precision measurements, the position of the reference feature must be determined with subpixel accuracy and has to be robust against noise influences.

Calibration: To enable a highly accurate reconstruction of specular surfaces, precise calibration is of essential importance for deflectometry. To triangulate surface points with sufficient accuracy, an intrinsic calibration of the camera and the monitor, as well as an extrinsic calibration of the measurement setup is mandatory. Due to the complex optical design of light field cameras, in this work, special emphasis is given to appropriate camera calibration.

Regularization: In order to find an unambiguous solution for the specular surface, additional information is needed. Hence, the special properties of the light field camera are to be used to resolve the ambiguity of the deflectometric measurement and to allow extracting the true surface normal from the one-dimensional solution manifold.

Reconstruction: The regularization provides an initial estimate of the surface. As deflectometry is a slope-measuring technique, the surface should be obtained by integrating the normal field. The light field camera is to be used to enable a robust reconstruction.

1.1 Contributions

The main contributions of this thesis are as follows.

- Deflectometry requires the coding of the reference monitor pixels. For this, phase-shift coding is used and different approaches for phase unwrapping are investigated. Since many approaches neglect the periodicity of the phase, modifications are proposed to improve the performance of existing unwrapping methods by using a circular mean operation and circular distances. Furthermore, a new probabilistic approach for temporal phase unwrapping is developed that uses circular statistics to model the multi-frequency phase-shift coding and enables an optimal reconstruction of the phase. The developed method respects the periodicity of the phase, simultaneously unwraps all phase measurements using maximum-likelihood estimation, allows for an easy frequency selection with a maximum uniqueness range of the unwrapping, and additionally, includes the estimation of the phase uncertainty into the overall unwrapping process. Moreover, the method is extended by considering the local pixel neighborhood resulting in a probabilistic approach for spatio-temporal phase unwrapping that outperforms state-of-the-art methods.
- Since light field cameras have a complex optical design, more sophisticated camera models are necessary. This work proposes to use a generic camera model, and a new approach for its calibration is developed in which the uncertainty of the calibration features is taken into account during optimization, leading to increased accuracy of the overall camera calibration. The problem is divided into two subproblems, a camera ray calibration and a reference target pose estimation, and to make the optimization feasible, alternating minimization is applied. Further, a closed-form least-squares solution for the ray calibration subproblem is presented, and the pose estimation subproblem is efficiently solved using a gradient descent optimization on the rotation manifold. All of this is achieved by minimizing a single objective function, where convergence is guaranteed. In addition, acceleration techniques are applied to obtain an almost quadratic convergence rate. Experimental

evaluations show that the proposed method outperforms standard calibration techniques and other generic approaches and yields a high-precision calibration.

- The reference monitor is modeled by a polynomial shape model and a refraction model, and it is shown how the model and the estimation of its parameters can be efficiently integrated into the generic calibration framework, which further increases the calibration accuracy.
- A new approach for light field reconstruction is developed using the generic camera calibration as a basis. The approach is completely generic and can be used to reconstruct light fields from arbitrary light field imaging systems, independent of whether the camera is based on microlenses, mirrors, or coded apertures, or whether it is realized by employing a camera array. Despite being estimated from a generic and unconstrained set of camera rays, the method outperforms state-of-the-art light field calibration approaches and yields rectified images with an accurate intrinsic calibration.
- Different approaches are developed to use the light field camera for the regularization of the deflectometric normal measurement. For partially specular surfaces, a classical light field depth estimation approach is used to obtain an initial estimate of the surface height. Furthermore, an approach is presented that estimates the distance to the reflected reference monitor and uses this to calculate the distance to the surface. For the reconstruction of fully specular freeform surfaces, a stereo deflectometry approach is adapted to implement a light field-based multi-view-deflectometry approach that allows triangulating of the surface.
- For the reconstruction of specular surfaces, a method is developed that fuses the depth estimates obtained through regularization with the deflectometrically measured surface normals. For this purpose, a variational fusion approach is adapted to account for the multi-view property of the light field camera, resulting in an improved reconstruction accuracy.

Some parts of this work have already been published elsewhere:

- The analysis of phase-shift coding and the methods for phase unwrapping from chapter 4 were published in [A7, A2, A1].
- The uncertainty-based calibration approach for the generic camera model from Sec. 5.2 and parts of the evaluation from Sec. 5.5 were published in [A5].
- Contributions to the generic light field reconstruction, as presented in chapter 6, were published in [A4, A3].
- Some aspects of the light field-based regularization techniques and the fusion of depth and normal estimates for surface reconstruction, as presented in chapter 7, were published in [A9, A6].

In contrast to these original publications, the content of this thesis has been changed considerably and the evaluation is much more detailed. In particular, not only partial aspects are examined. Instead, the interaction of individual components with each other is investigated, and the influence of the entire processing chain on the performance of the final result is examined.

1.2 Overview

The remainder of this thesis is structured as follows. Starting with chapter 2, basic mathematical concepts used throughout this work are presented. Chapter 3 provides the theory of light fields, light field imaging, and its applications. Furthermore, it explains the working principles of deflectometry, presents its difficulties, and formulates the steps required for light field-based specular surface reconstruction.

In chapter 4 the first step of the deflectometric measurement pipeline is analyzed, *i.e.*, the registration of camera pixels with features on a reference monitor. The principles of phase-shift coding are introduced, the state of the art in the field of temporal phase unwrapping is reviewed, suggestions for improvement are made, and as the main content, a new probabilistic approach to spatio-temporal phase unwrapping is presented. The chapter

concludes with an extensive comparison of the proposed methods with state-of-the-art methods.

Chapter 5 presents the calibration of the entire deflectometric measurement setup. Starting from the basic principles of camera calibration, the motivation behind the use of more advanced generic camera models is introduced, and an alternating minimization-based approach for calibration is presented, taking into account the uncertainty of calibration features that were obtained through phase-shift coding. After this, the modeling of the reference monitor is explained, and it is demonstrated how estimating its parameters can be integrated into the generic calibration framework. Furthermore, it is shown how the generic camera model can be utilized to perform the extrinsic calibration of the deflectometric measurement setup.

Subsequently, in chapter 6, the results of the generic camera calibration are reused to decode light fields from raw camera data. With this, the inherent 4D topological ray-space of the light field is reconstructed, preserving both the information of the observed scene and the geometric structure of the light field by adequate rectification and calibration. Further, different resampling strategies are discussed and the proposed method is compared to state-of-the-art light field calibration methods.

Eventually, chapter 7 demonstrates how light field cameras can be efficiently combined with deflectometry. Possibilities for a light field-based regularization are proposed, which can solve the ambiguity of the surface normal estimation. A variational surface reconstruction approach is presented, which fuses the regularization points with the deflectometrically measured surface normals and enables high-precision reconstruction. Furthermore, different surfaces are investigated and several aspects of the entire deflectometric measurement chain are examined for their influence on the surface reconstruction.

Finally, chapter 8 summarizes the presented work and draws conclusions, providing further insights into future research possibilities.

2 Preliminaries

This chapter introduces the basic mathematical principles used in this work. These include useful operators, the mathematical parameterization of rotations, lines, and surfaces in 3D space, and optimization techniques. The purpose of this chapter is to provide a general list of tools needed for this work. All information is gathered here to avoid impeding the flow of reading in later chapters. The following is therefore primarily intended as a reference.

2.1 Operators

Reshape Operators

The vec -operator vectorizes a matrix by stacking its columns:

$$\text{vec}(\mathbf{B}) := \begin{pmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_M \end{pmatrix}, \text{ with } \mathbf{B} = (\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_M) \quad (2.1)$$

where $\mathbf{B} \in \mathbb{R}^{L \times M}$, $\mathbf{b}_i \in \mathbb{R}^L$ and $\text{vec}(\mathbf{B}) \in \mathbb{R}^{LM}$.

The mat -operator is the inverse of the vec -operator, and reshapes a vectorized matrix back to its original form:

$$\text{mat}(\mathbf{b}) := \mathbf{B}, \text{ where } \mathbf{b} = \text{vec}(\mathbf{B}). \quad (2.2)$$

The vec -operator is compatible with the Kronecker product \otimes . With $\mathbf{A} \in \mathbb{R}^{K \times L}$, $\mathbf{B} \in \mathbb{R}^{L \times M}$, $\mathbf{C} \in \mathbb{R}^{M \times N}$, a useful equation can be derived [63]:

$$\text{vec}(\mathbf{ABC}) = (\mathbf{C}^T \otimes \mathbf{A})\text{vec}(\mathbf{B}), \quad (2.3)$$

where $(\mathbf{C}^T \otimes \mathbf{A}) \in \mathbb{R}^{KN \times LM}$ and $\text{vec}(\mathbf{B}) \in \mathbb{R}^{LM}$.

Skew-Operator

The cross-product between vectors can be formulated using the skew-operator $[\cdot]_{\times}$. For $\xi, \mu \in \mathbb{R}^3$ the skew-operator is defined as follows:

$$[\xi]_{\times} := \begin{bmatrix} 0 & -\xi_3 & \xi_2 \\ \xi_3 & 0 & -\xi_1 \\ -\xi_2 & \xi_1 & 0 \end{bmatrix}. \quad (2.4)$$

With it, the useful relations

$$\xi \times \mu = [\xi]_{\times} \mu = [\mu]_{\times}^T \xi = -\mu \times \xi \quad (2.5)$$

can be formulated. Applying the vec -operator on the skew-operator can be formulated as a matrix-vector product:

$$\text{vec}([\xi]_{\times}) = \mathbf{Z} \xi, \quad (2.6)$$

$$\mathbf{Z} = [\text{vec}([\mathbf{e}_1]_{\times}), \text{vec}([\mathbf{e}_2]_{\times}), \text{vec}([\mathbf{e}_3]_{\times})], \quad (2.7)$$

with the unit basis vectors $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$.

Directional Derivative

Let \mathcal{M} be a smooth submanifold of a Euclidean space and \mathbf{p} a point of \mathcal{M} . Let f be a function defined in a neighborhood of \mathbf{p} that is differentiable at \mathbf{p} . With the tangent vector ξ to \mathcal{M} at \mathbf{p} , the directional derivative of f along ξ , can be defined. Given a curve γ on \mathcal{M} with $\gamma(0) = \mathbf{p}$ and $\dot{\gamma}(0) = \xi$, the directional derivative is defined by [1]

$$D_{\xi} f(\mathbf{p}) := \partial_{\varepsilon} f(\gamma(\varepsilon))|_{\varepsilon=0}. \quad (2.8)$$

2.2 Rotation Parametrization

Rotations in 3D space have three degrees of freedom. There are different parametrizations, which contain more or less redundant information, and which are subject to more or less constraints [188]. Depending on the application for which a mathematical description of rotations is required, different parametrizations can be advantageous. In this work, rotations are represented throughout by rotation matrices.

Rotation Matrix

Rotation matrices are elements of the special orthogonal group in three dimensions $\mathbf{R} \in \text{SO}(3) \subset \mathbb{R}^{3 \times 3}$, which are subject to several constraints:

$$\text{SO}(3) = \{\mathbf{R} \in \mathbb{R}^{3 \times 3} \mid \mathbf{R}^T \mathbf{R} = \mathbf{I}, \det(\mathbf{R}) = 1\} . \quad (2.9)$$

A rotation matrix is described with nine parameters

$$\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} = (\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3) , \quad (2.10)$$

where $\|\mathbf{r}_n\| = 1$ for $n = 1, 2, 3$. The transposed rotation matrix is its own inverse $\mathbf{R}^T = \mathbf{R}^{-1}$, and the column vectors $\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3$ span the coordinate system that the rotation matrix transforms to.

Local Parametrization of SO(3)

Rotation matrices are very intuitive because the rotation of a 3D point can be realized by simple matrix-vector multiplication. If rotations are needed for parameter estimation or optimization, the highly redundant rotation matrices are only of limited use due to the many constraints. If used in optimization, derivatives often have to be calculated. However, the simple calculation of derivatives of individual parameters does not lead to meaningful results, since rotation matrices are defined on the Riemannian manifold $\text{SO}(3)$ and this property is lost if not handled correctly [24]. Derivatives must therefore be calculated directly on the manifold [81, 138, 176].

The smooth and differentiable Riemannian manifold $\text{SO}(3)$ is a finite-dimensional Lie group [189]. Every matrix Lie group is associated with a Lie algebra. The corresponding Lie algebra $\mathfrak{so}(3)$ is the set of all 3×3 skew-symmetric matrices

$$\mathfrak{so}(3) = \{\boldsymbol{\Omega} = [\boldsymbol{\xi}]_{\times} \in \mathbb{R}^{3 \times 3} \mid \boldsymbol{\xi} \in \mathbb{R}^3\} , \quad (2.11)$$

which is the tangent space of the Lie group at the identity element [24].

The mapping from any element $[\boldsymbol{\xi}]_{\times} \in \mathfrak{so}(3)$ to $\mathbf{R} \in \text{SO}(3)$ is called the exponential map $\mathbf{R} = \text{Exp}([\boldsymbol{\xi}]_{\times})$ and is defined using the standard

matrix exponential series. It can be calculated in closed form using the well known Rodrigues rotation formula [125]:

$$\text{Exp}([\xi]_{\times}) := e^{[\xi]_{\times}} = \mathbf{I} + \frac{[\xi]_{\times}}{\|\xi\|} \sin(\|\xi\|) + \frac{[\xi]_{\times}^2}{\|\xi\|^2} (1 - \cos(\|\xi\|)) . \quad (2.12)$$

The reverse map from the Lie group to the Lie algebra $[\xi]_{\times} = \text{Log}(\mathbf{R})$ is called the logarithmic map [125]:

$$\text{Log}(\mathbf{R}) := \frac{\theta(\mathbf{R} - \mathbf{R}^T)}{2 \sin(\theta)} , \text{ with } \theta = \cos^{-1} \left(\frac{\text{trace}(\mathbf{R}) - 1}{2} \right) . \quad (2.13)$$

Therefore, one can find a smooth parametrization $g_{\mathbf{R}}(\xi) = \text{Exp}([\xi]_{\times}) \mathbf{R}$ of the $\text{SO}(3)$ manifold in a local neighborhood of \mathbf{R} , which is differentiable with respect to $\xi \in \mathbb{R}^3$ using the tangent space.

2.3 Line Parametrization and Plücker Coordinates

In 6D-Plücker-space a Plücker-line $\mathbf{l} \in \mathbb{P}^6$ is defined by its direction vector $\mathbf{d} \in \mathbb{R}^3$ and its moment vector $\mathbf{m} \in \mathbb{R}^3$ [186, 207]. A line in 3D-space has four degrees of freedom, therefore two constraints apply to the Plücker-line:

$$\mathbb{P}^6 = \left\{ \begin{pmatrix} \mathbf{d} \\ \mathbf{m} \end{pmatrix} \middle| \mathbf{d}, \mathbf{m} \in \mathbb{R}^3, \mathbf{d}^T \mathbf{m} = 0, \|\mathbf{d}\| = 1 \right\} . \quad (2.14)$$

The moment vector can be calculated with $\mathbf{m} = \mathbf{p}_1 \times \mathbf{d}$, where $\mathbf{p}_1 \in \mathbb{R}^3$ is an arbitrary point on the line \mathbf{l} , see figure 2.1. The moment vector stands perpendicular to the line and its norm $\|\mathbf{m}\|$ corresponds to the Euclidean distance of the line to the origin. Given two points $\mathbf{p}_1, \mathbf{p}_2 \in \mathbb{R}^3$, the Plücker-line $\mathbf{l}^T = (\mathbf{d}^T, \mathbf{m}^T)$ traversing both points can be calculated:

$$\mathbf{d} = \frac{\mathbf{p}_1 - \mathbf{p}_2}{\|\mathbf{p}_1 - \mathbf{p}_2\|} , \quad (2.15)$$

$$\mathbf{m} = \mathbf{p}_1 \times \mathbf{d} = \mathbf{p}_2 \times \mathbf{d} . \quad (2.16)$$

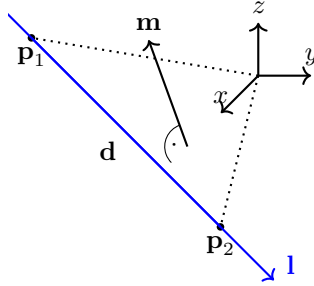


Figure 2.1 A Plücker-line \mathbf{l} is defined by its direction vector \mathbf{d} and moment vector \mathbf{m} . It can be calculated from two points \mathbf{p}_1 and \mathbf{p}_2 on the line.

A rotation and translation of the line in 3D-space is achieved using simple matrix operations [12]:

$$\mathbf{l}' = \mathbf{R}_1 \mathbf{l} = \begin{pmatrix} \mathbf{R} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{pmatrix} \mathbf{l}, \quad (2.17)$$

$$\mathbf{l}' = \mathbf{T}_1 \mathbf{l} = \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ [\mathbf{t}]_{\times} & \mathbf{I} \end{pmatrix} \mathbf{l}, \quad (2.18)$$

where $\mathbf{R} \in \text{SO}(3)$, $\mathbf{t} \in \mathbb{R}^3$ and $[\cdot]_{\times}$ are the rotation matrix, the translation vector and the skew operator, respectively. The Euclidean distance $d(\mathbf{l}, \mathbf{p})$ of a line \mathbf{l} to an arbitrary point $\mathbf{p} \in \mathbb{R}^3$ is defined as the distance to the closest point on the line. It is found by translating the origin of the coordinate system into the point \mathbf{p}

$$\mathbf{l}' = \begin{pmatrix} \mathbf{d}' \\ \mathbf{m}' \end{pmatrix} = \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ [-\mathbf{p}]_{\times} & \mathbf{I} \end{pmatrix} \mathbf{l} = \begin{pmatrix} \mathbf{d} \\ -[\mathbf{p}]_{\times} \mathbf{d} + \mathbf{m} \end{pmatrix} \quad (2.19)$$

and by calculating the distance between the translated line and the new origin:

$$d(\mathbf{l}, \mathbf{p}) = d(\mathbf{l}', \mathbf{0}) = \|\mathbf{m}'\| = \|\mathbf{p} \times \mathbf{d} - \mathbf{m}\|. \quad (2.20)$$

2.4 Surface Parametrization

Surfaces are represented in this work by point clouds or, when needed, by a two-dimensional function

$$\begin{aligned} z : \mathbb{R}^2 &\rightarrow \mathbb{R}, \\ (s, t) &\mapsto z(s, t). \end{aligned} \quad (2.21)$$

where the set of s, t values defines the topological relationship (e.g., between camera pixels), and $z(s, t)$ represents the corresponding depth or height value. The surface is hereby defined by discrete values or a continuous implicit parametric description.

Relation between Surface Normal and Gradient

Let s, t be the image coordinates of a surface, $\mathbf{n}(s, t)$ the corresponding surface normal, and $\mathbf{x}(s, t) = (x(s, t), y(s, t), z(s, t))^T$ a surface point. Calculating the surface gradient with respect to the image coordinates now depends on the model of projection [163].

With an orthographic projection a 3D point is projected orthogonally onto the image plane. The image coordinates equal the point coordinates:

$$x(s, t) = s, \quad (2.22)$$

$$y(s, t) = t. \quad (2.23)$$

The cross product of the 3D point's partial derivatives is normal to the surface:

$$\partial_s \mathbf{x} \times \partial_t \mathbf{x} \sim \mathbf{n}. \quad (2.24)$$

By normalizing this, and choosing the sign so that \mathbf{n} points toward the camera, one obtains

$$\mathbf{n} = \frac{1}{\sqrt{1 + \|\nabla z\|^2}} \begin{pmatrix} \partial_s z \\ \partial_t z \\ -1 \end{pmatrix}, \quad (2.25)$$

where $\nabla z = (\partial_s z, \partial_t z)^T$ denotes the gradient of the depth map z . Solving (2.25) for the surface gradient then yields

$$\mathbf{g} := \nabla z = \begin{pmatrix} \partial_s z \\ \partial_t z \end{pmatrix} = -\frac{1}{n_3} \begin{pmatrix} n_1 \\ n_2 \end{pmatrix}. \quad (2.26)$$

With a perspective projection, the projected coordinates now depend on the depth and the focal length f of the used camera [163]:

$$x(s, t) = \frac{z(s, t)}{f} s, \quad (2.27)$$

$$y(s, t) = \frac{z(s, t)}{f} t. \quad (2.28)$$

The cross product of the 3D point's partial derivatives is normal to the surface and parallel to the normal vector, implying $\partial_s \mathbf{x} \times \partial_t \mathbf{x} \times \mathbf{n} = \mathbf{0}$, which results in the equation system

$$\begin{aligned} 0 &= f n_3 \partial_s z + n_1 [z + s \partial_s z + t \partial_t z], \\ 0 &= f n_3 \partial_t z + n_2 [z + s \partial_s z + t \partial_t z], \\ 0 &= n_2 \partial_s z - n_1 \partial_t z. \end{aligned} \quad (2.29)$$

Knowing $z > 0$ holds for the depth map and substituting $\bar{z} = \ln(z)$ makes (2.29) linear in the partial derivatives $\partial_u \bar{z}$ and $\partial_v \bar{z}$:

$$\begin{aligned} 0 &= [n_3 f + n_1 s] \partial_s \bar{z} + n_1 t \partial_t \bar{z} + n_1, \\ 0 &= [n_3 f + n_2 t] \partial_t \bar{z} + n_2 s \partial_s \bar{z} + n_2, \\ 0 &= n_2 \partial_s \bar{z} - n_1 \partial_t \bar{z}. \end{aligned} \quad (2.30)$$

This can then be easily inverted, providing a formula for the surface gradient of the substitute depth map \bar{z} :

$$\bar{\mathbf{g}} := \nabla \bar{z} = \begin{pmatrix} \partial_s \bar{z} \\ \partial_t \bar{z} \end{pmatrix} = -\frac{1}{s n_1 + t n_2 + f n_3} \begin{pmatrix} n_1 \\ n_2 \end{pmatrix}. \quad (2.31)$$

2.5 Primal-Dual Optimization

For variational optimization, often the primal-dual formalism is used to find efficient optimization algorithms that allow for a smooth minimization of non-smooth functions [34].

Let \mathcal{X}, \mathcal{Y} be two finite-dimensional real vector spaces and let the general optimization problem be of the form

$$\min_{\mathbf{x} \in \mathcal{X}} F(\mathbf{K}\mathbf{x}) + G(\mathbf{x}), \quad (2.32)$$

where $\mathbf{K} : \mathcal{X} \rightarrow \mathcal{Y}$ is a continuous linear operator, $F : \mathcal{Y} \rightarrow \mathbb{R}^+$, $G : \mathcal{X} \rightarrow \mathbb{R}^+$ are convex functions, while F can be discontinuous. The primal-dual formulation of this is the convex-concave saddle point problem [25]

$$\min_{\mathbf{x} \in \mathcal{X}} \max_{\mathbf{y} \in \mathcal{Y}} \langle \mathbf{K}\mathbf{x}, \mathbf{y} \rangle + G(\mathbf{x}) - F^*(\mathbf{y}), \quad (2.33)$$

where $\langle \cdot, \cdot \rangle$ is an inner product, \mathbf{x} is considered the primal variable, \mathbf{y} the dual variable, and F^* denotes the convex conjugate of the function F :

$$F^*(\mathbf{y}) = \sup_{\mathbf{x} \in \mathcal{X}} \{ \langle \mathbf{y}, \mathbf{x} \rangle - F(\mathbf{x}) \}. \quad (2.34)$$

Independent of the convexity of F , the convex conjugate is always a convex function. The saddle point optimization problem can be efficiently solved in an alternating manner using primal-dual algorithms [34]:

$$\mathbf{y}^{(n+1)} = \text{prox}_{\sigma F^*} \left(\mathbf{y}^{(n)} + \sigma \mathbf{K} \bar{\mathbf{x}}^{(n)} \right), \quad (2.35)$$

$$\mathbf{x}^{(n+1)} = \text{prox}_{\tau G} \left(\mathbf{x}^k - \tau \mathbf{K}^* \mathbf{y}^k \right), \quad (2.36)$$

$$\bar{\mathbf{x}}^{(n+1)} = \mathbf{x}^{(n+1)} + \theta \left(\mathbf{x}^{(n+1)} - \mathbf{x}^k \right), \quad (2.37)$$

where τ, σ, θ are parameters and \mathbf{K}^* is the adjoint of the operator \mathbf{K} . The primal variable \mathbf{x} is updated in each iteration n with a proximal descend, the dual variable \mathbf{y} is updated with a proximal ascend, and a final extrapolation step increases the convergence rate. The proximal operators can be formulated through optimization of an independent subproblem:

$$\text{prox}_{\tau G}(\mathbf{x}) = \arg \min_{\mathbf{x}'} \left\{ \frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\tau} + G(\mathbf{x}') \right\}. \quad (2.38)$$

The advantage of the primal-dual algorithms is that the difficult optimization problem (2.32) can be iteratively solved, in which only the proximal operators need to be evaluated, where in many cases an analytic solution for the subproblems can be provided [146].

3 Background

3.1 Deflectometry

Specular surfaces can be found in numerous areas of industrial production. For instance, they appear in lacquered body parts of the automotive industry, in entertainment products, in glazed ceramics, or in the production of high-precision mirror optics, such as those used in telescopes. Depending on the degree of specularity, an observation will reveal image features composed of a superposition of direct surface features and features from the reflected image of the environment. Obtaining 3D information about physical objects is a significant application of automated visual inspection methods. However, many of these methods fail when examining fully specular objects, especially triangulation-based methods such as stereo vision or fringe projection profilometry. The reason for this is that, in contrast to diffuse reflection, an observer does not see the surface itself, but the distorted mirror image of the surroundings. The specular surface is practically invisible to the observer. Automatic visual inspection, especially 3D measurement, therefore is a major challenge.

While a human observer can intuitively make assumptions about the surface by watching the distortion, various computer vision techniques try to imitate this principle, *e.g., shape from specular reflection* and *shape from distortion* [11]. A certain subclass of these are the so-called deflectometric methods. The measurement setup consists here of a camera and an active illumination source, *e.g.,* a commercially available monitor. By illuminating with a known reference pattern, information about the surface can be obtained from the observed distortions. In detail, *deflectometry* makes it possible to obtain highly precise slope information of the surface, which can be used for 3D reconstruction or defect detection. The advantages of deflectometry are that it is very robust, can be realized with inexpensive hardware, and the measurement sensitivity is limited geometrically by

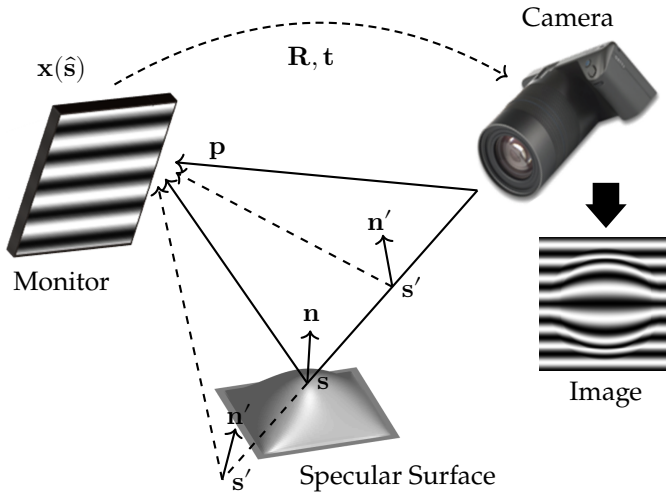


Figure 3.1 Deflectometric measurement principle: The camera observes distorted reference patterns as reflection in the surface. Knowing the reference point, a surface normal can be calculated for every point on a camera ray.

the resolution of the camera sensor and the extent of the measurement setup. This makes it interesting for many industrial applications.

3.1.1 Measurement Principle

The most basic experimental setup for a deflectometric measurement is illustrated in figure 3.1. It consists of three components: an illumination source displaying structured patterns, a specular object under test, and a camera. For the light source, standard LCD monitors are usually used that can be actively controlled, or reference patterns are projected onto a canvas by means of a projector. The reference shows a pattern or a series of patterns, which are then reflected on the examined specular object. In deflectometry, the specular surface itself is part of the system and is located in the optical path between the illumination and the camera. The reference pattern is therefore distorted by the curvature of the surface, and the resulting warped pattern can be imaged with a conventional digital camera. Assuming each ray is reflected only once, which is true

for many technical surfaces, and since the object is specularly reflective, a camera pixel sees either exactly one position on the screen or none.

In a camera-fixed coordinate system, starting from a camera pixel, a vision ray can be constructed with direction $\hat{s} \in \mathbb{S}^2$ starting from the optical center of the camera. The ray hits the surface in the point $s = \rho\hat{s}$, with $\|\hat{s}\| = 1$. At the surface, the ray is reflected, hits the reference monitor, and observes a feature $x(\hat{s})$ in the monitor plane. If in a fully calibrated system the transformation of monitor coordinates to the camera coordinate system is known, the position of the observed monitor feature relative to the camera can be calculated:

$$\mathbf{p} = \mathbf{R}\mathbf{x} + \mathbf{t}. \quad (3.1)$$

Using the law of reflection, the surface normal \mathbf{n} of the observed point can then be specified as the angle bisector between the camera ray \hat{s} and the reflected ray \hat{s}_r :

$$\mathbf{n}(\rho) = \hat{s}_r - \hat{s} = \frac{\mathbf{p} - \mathbf{s}}{\|\mathbf{p} - \mathbf{s}\|} - \frac{\mathbf{s}}{\|\mathbf{s}\|} = \frac{\mathbf{p} - \rho\hat{s}}{\|\mathbf{p} - \rho\hat{s}\|} - \hat{s}. \quad (3.2)$$

The integration of the normal field of all camera pixels finally yields the reconstruction of the investigated surface.

However, a problem arises here, because in general, the length ρ of the vector \mathbf{s} is unknown in (3.2). This means that a one-parametric set of hypothetical surface normals can be calculated for each camera ray, which in turn leads to an ambiguity of the surface estimation. More precisely, a surface normal can only be calculated correctly if the corresponding surface point is already known, and the surface can only be reconstructed if the surface normals are provided. To resolve the ambiguity of the deflectometric measurement, additional regularizing information is required. In principle, it would suffice to measure only one point of the surface and to reconstruct the surface from the normal field starting from this point by assuming a continuous surface [11]. However, if more samples are available, this can help to reduce the influence of an uncertain and noisy measurement of a single surface point.

3.1.2 Related Works

Deflectometry has a long history in computer vision and optical metrology. Among the earliest work, Sanderson *et al.* [174] proposed a structured highlight illumination approach using an array of point light sources to illuminate a specular surface, and they estimated surface orientation using a stereo camera. The first promising results for optical metrology were demonstrated in the work of Petz and Ritter [151] and Petz and Tutsch [152, 153]. They proposed reflectance grating photogrammetry for the measurement of specular surfaces by using a linear positioning unit to move a flat reference structure into different positions from which the illumination direction is derived. By applying the triangulation principle between the camera rays and known illumination direction, they then determine point-wise the absolute 3D object coordinates with high precision. Knauer *et al.* [103] analyzed the investigation of specular freeform surfaces through phase-shift coding and introduced the term *phase measuring deflectometry* for the first time. They further described many aspects, *e.g.*, the measurement principle, the physical limits of the method, and the calibration of the system components. Bothe *et al.* [22] gave practical demonstrations of their fringe reflection technique, which allowed nondestructive testing of specular surfaces and high-resolution 3D shape measurement. And thus, deflectometry was promoted as a novel technique for the measurement of specular freeform surfaces.

Applications

Since for industrial applications often only quality assurance or defect detection is of interest, many pure inspection methods exist. Häusler *et al.* [74] proposed a microscopic PMD system with nanometer sensitivity for local surface features. Xiao *et al.* [228] used deflectometry to measure the 3D shape of aspherical mirrors. Olesch *et al.* [142] used deflectometry for large-scale estimation of telescope mirrors. Häusler *et al.* [75] and Faber *et al.* [51] compared deflectometry with interferometry and describe the advantages and disadvantages. Werling and Beyerer [220] proposed inverse patterns that are computed in advance for known test objects and that can be used for fast and robust defect detection on specular objects. Su *et al.* [192] investigated deflectometry with an infrared source to

analyze rough optical surfaces. Höfer *et al.* [83] and Höfer [82] presented approaches that allow coding of the reference patterns in the infrared spectrum, and thus render infrared deflectometry industrially useful.

Regularization

The deflectometric normal measurement is inherently ambiguous, thus, additional information is needed for 3D reconstruction. Therefore, several approaches for regularization exist.

Li *et al.* [113] use an additional confocal white-light distance sensor to precisely determine a single surface point from which the surface can be reconstructed. Huang *et al.* [87] use an external laser tracker to precisely measure the system setup and mirror surface position, and compare it with a virtual system setup, which subsequently yields a high precision reconstruction.

When additional assumptions are made about the surface, the reconstruction can be simplified and a solution approximated. By neglecting higher-order surface properties, the reconstruction task can be reduced to a finite-dimensional parameter estimation problem, which in general has a unique solution [9]. Liu *et al.* [118] show that the surface can be reconstructed uniquely under certain conditions if it is at least twice continuously differentiable. Pak [143] adopts this approach and simplifies the mathematical description. Liang *et al.* [114] characterize the surface locally as a low-dimensional model and build their approach on the work of Savarese *et al.* [177]. Huang *et al.* [85] describe the surface using a global model, and they find the surface through parameter optimization.

Various methods exist to reconstruct the direction of the illumination utilizing a multi-monitor approach. Here, the monitor can be moved with a linear positioning unit [152, 153], or the approach can be implemented without mechanical movements by using a beam splitter and two separate monitors [120, 247]. In this context, Han *et al.* [71] present an idea that can reconstruct the surface even with an uncalibrated camera model and unknown monitor poses. Similarly, a directed illumination can also be realized with telecentric optics, which enables triangulating the surface [184, 237].

When attempting to apply classical stereo vision to specular surfaces, initially there is the difficulty that only virtual features can be captured

in the two camera images (*cf.* Sec. 7.1.2). However, *specular stereo* can be achieved by correlating the normal vector fields induced by two measurements, which indirectly enables surface triangulation. For this purpose, starting from the ambiguous normal vector field, Bhat and Nayar [17] seek the simultaneous solution of two partial differential equations, one for each viewpoint. Bonfort and Sturm [21] use a correlation measure for point-wise reconstruction by voxel carving. Balzer *et al.* [10] extend the principle to measure large objects by using multi-view specular stereo.

The limit case of the stereo approach is represented by *specular flow*. It is assumed that the movement between image configurations is so small that the correspondence between image and scene points is maintained across two images. Roth and Black [172] combine diffuse and specular flow to reconstruct partially specular surfaces. Balzer [9] derives model equations of specular flow that can also describe nonlinear camera motions. Adato *et al.* [2] provide a solution for shape from specular flow, which makes it possible to reconstruct the surface by observing the specular flow induced by an unknown environment motion field. Pak [144] derives a simple relation between specular flow and the Gaussian curvature of specular surfaces. However, the method has a practical disadvantage: no coded illumination can be used because the camera has to be moved continuously for specular flow.

Reconstruction

Although regularization provides a rough estimate of the surface, the advantages of deflectometry are that it can determine the local slope of the surface very precisely, *i.e.*, the measurement of the surface normals is generally significantly more precise than the direct measurement of surface points. Based on the unambiguous surface normals obtained from the regularization, the surface can be reconstructed.

Various works exist that describe the surface using a two-dimensional polynomial and convert the reconstruction into a parameter estimation problem [85]. For this, depending on the shape of the specular object, different surface models are used, *e.g.*, Zernike polynomials, radial basis functions, or Forbes polynomials [50, 85, 166].

Other approaches consider the reconstruction problem as normal integration or gradient integration. In principle, there are two concepts

for this. Local methods integrate the surface along predetermined paths. Horbach and Dang [84] propagate the regularization information by region growing starting from at least one known surface point to integrate the normal field. Neighboring surface normals can be computed by assuming the continuous differentiability of the surface, which allows a local regularization. However, in doing so, they also propagate the measurement error along the path leading to a global shape deviation [50]. Since the normal field is typically corrupted by noise, it is therefore seldom integrable and curl-free. Therefore, the error of the integration depends on the chosen path. For this reason, variational approaches are often used as global methods, where only the integrable part of the normal field is considered, and the integration task is formulated as an energy minimization problem [163]. Since the integration of surface normals also occurs in many other applications (*e.g.*, photometry, profilometry), there exists much literature on the subject. Chang *et al.* [35] use level set methods to integrate a multi-view normal field and apply this to photometric stereo images. In the case of deflectometry, Balzer *et al.* [10] use multi-view regularization to obtain an initial surface estimate, and they refine this by integrating the normal field. This is done by iteratively solving a Poisson equation using finite-elements analysis, updating the reconstructed normals and the measured normal field concurrently. Quéau and Durou [162] explore edge-preserving integration of normal fields by examining different energy functionals for the reconstruction of discontinuous surfaces. Quéau *et al.* [164] present several total variation-like integration approaches where surface normals and depth estimates can be fused into one surface. Antensteiner *et al.* [8] compare different algorithms that fuse depth values with gradient estimates, with application to light field photometric stereo.

While there is still other related work relevant for deflectometry, *e.g.*, the coding of the structured illumination and the calibration of the measurement system, they are not discussed here but later in their associated chapters, see Ch. 4 and Ch. 5. For more details on deflectometry, its applications, further regularization and reconstruction techniques, the reader is referred to the comprehensive reviews in the literature [11, 86, 163, 222].

3.2 Light Fields

The light propagating through space contains a variety of information. Within the field of geometrical optics, the theoretical background for a description of this propagation is provided by the *plenoptic function* that assigns a radiance value to the light rays present in a physical space. It assumes that the usual 3D space is traversed by light propagating in all directions, and the light may be blocked, attenuated, or scattered. To account for all possible variations of light, the plenoptic function takes a seven-dimensional description $P(\mathbf{x}, \boldsymbol{\theta}, \lambda, \tau) : \mathbb{R}^7 \rightarrow \mathbb{R}$. Arbitrary radiance values can be assigned at any location in space $\mathbf{x} \in \mathbb{R}^3$, for any possible directional angle $\boldsymbol{\theta} \in \mathbb{R}^2$, any wavelength λ , and any time τ . While the plenoptic function is mainly of conceptual interest in this work, recently, it found applications in the field of scene reconstruction and novel view synthesis [129, 236].

In contrast, light fields have a more practical meaning, since they allow the description of imaging systems in which only the rays that reach the camera sensor are relevant. By introducing additional constraints, the light field can be derived from the plenoptic function [92]. If only single points in time are considered or if the light is integrated over the exposure time, the temporal dimension τ of the plenoptic function can be omitted. The integration over the spectral sensitivity of the camera pixels eliminates the spectral dimension λ of the plenoptic function. Thus, the light field is considered monochromatic. However, in this work, color or more abstract coded information may be assigned to the rays, although this will not be implicitly stated. The most important reduction of dimensions is achieved by the so-called free space assumption [110]. In homogeneous media that are free of occluders, the radiance along a ray is constant. Hence, the spatial dependency of the plenoptic function can be reduced by one dimension. Moon and Spencer [132] called the resulting function *photoc field*, while in the field of computer graphics it is titled *4D light field* [110] or *Lumigraph* [64]. Formally, the 4D light field $L(u, v, s, t)$ is defined as the radiance along light rays in an empty space, where the coordinates (u, v, s, t) correspond to a certain parametrization of the spatial and angular dependencies of the light field. The array of rays in a light field can be modeled in different ways. The most commonly used parametrization is the two-plane parametrization, where a light

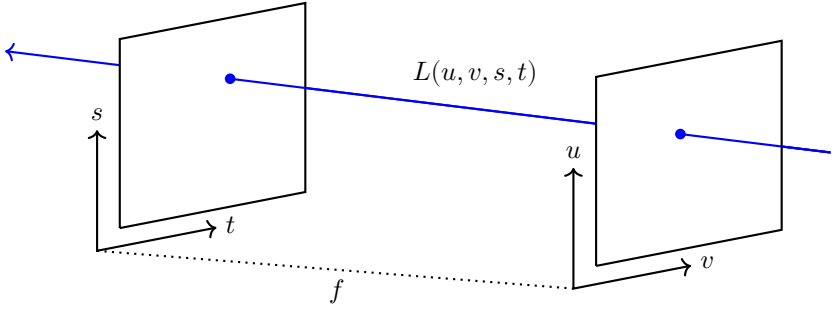


Figure 3.2 Two-plane parametrization of the 4D light field: A light ray is described by the coordinates of intersection with two parallel planes.

ray is uniquely described by the intersections of two parallel planes with angular coordinate (u, v) , spatial coordinates (s, t) , and with distance f between both planes, see figure 3.2. This may not represent all rays, for example, rays parallel to the two planes, provided the planes are parallel to each other. The advantage, however, is that its description is closely related to the analytical geometry of perspective imaging in optical systems.

A simple way to visualize the two-plane parametrization of the light field $L(u, v, s, t)$ is to imagine it as a discrete collection of many perspective images of the s, t -plane, each of which is taken from a different observation position in the u, v -plane with a virtual camera. Hence, for each fixed angular coordinate (u_0, v_0) a two-dimensional slice can be extracted from the light field, which in the following is called a subaperture image (SAI):

$$\text{SAI}_{u_0 v_0}(s, t) = L(u_0, v_0, s, t), \quad (3.3)$$

where each SAI resembles a conventional image. By fixing an angular coordinate and the spatial coordinate whose axis is parallel to that coordinate, a so-called epipolar plane image (EPI) is obtained:

$$\text{EPI}_{u_0 s_0}(v, t) = L(u_0, v, s_0, t), \quad (3.4)$$

$$\text{EPI}_{v_0 t_0}(u, s) = L(u, v_0, s, t_0), \quad (3.5)$$



Figure 3.3 Interpretation of the light field as a camera array. Each SAI represents a “virtual” camera that is slightly shifted with respect to the other cameras. The dashed lines indicate the coordinates of the extracted EPIs.

where, depending on which coordinates are fixed, a horizontal or vertical EPI is extracted. Figure 3.3 shows an example light field as an array of virtual cameras that are slightly shifted against each other, as well as a horizontal and a vertical EPI. Due to the change of perspective for each angular coordinate, the EPIs show lines of different slopes, whose orientation provides information about the depth of the observed scene points.

3.2.1 Light Field Acquisition

The easiest way to sample the continuous light field $L(u, v, s, t)$ is to use a mechanical gantry to place a conventional camera at different positions in the u, v -plane and capture the scene [205]. Of course, instead of a time-sequential capture, this can be efficiently implemented hardware-parallel using multi-camera arrays to allow obtaining high-resolution light fields [233]. While camera arrays can be miniaturized and a configuration of specialized camera modules can be assembled, building and maintaining camera arrays is costly and cumbersome.

In contrast, single-shot light field cameras have been proposed that image the light field through a single main lens and encode the four-dimensional information onto a two-dimensional camera sensor. The most commonly used designs for such light field acquisition devices are microlens-based light field cameras. While the basic idea of such cameras was already described by Lippmann [116] as early as 1908, only modern computing power and advances in the fabrication of microscopic structures made commercialization possible. The first design of a light field camera was introduced much later in 1992 by Adelson and Wang [3], who called it *plenoptic camera*. And one of the first hand-held prototypes was built by Ng *et al.* [137], which was then commercialized by Lytro Inc. The camera's layout is similar to that of a conventional camera with the essential difference that an array of microscopically sized lenses is placed in front of the sensor. By adding this microlens array (MLA), it becomes possible to capture a section of the 4D light field $L(u, v, s, t)$ of a scene and encode it onto the 2D sensor. In particular, there are different designs.

When the distance between the MLA and the sensor corresponds to the focal length of the microlenses, the camera is an *unfocused plenoptic camera*, see figure 3.4. The coordinates of the light field's two-plane parametrization are represented here by the s, t - and u, v -coordinates, whereby s, t define the position of a microlens in front of the sensor, and thus, they encode the spatial dimension of the light field. Hence, they can be interpreted as macro pixels. The u, v -coordinates define the position within the microlens relative to its center and in this way, they implicitly provide information on where a light ray has passed through the main lens. They represent the angular information of the light field. Since the microlenses are relatively small, their size is usually below

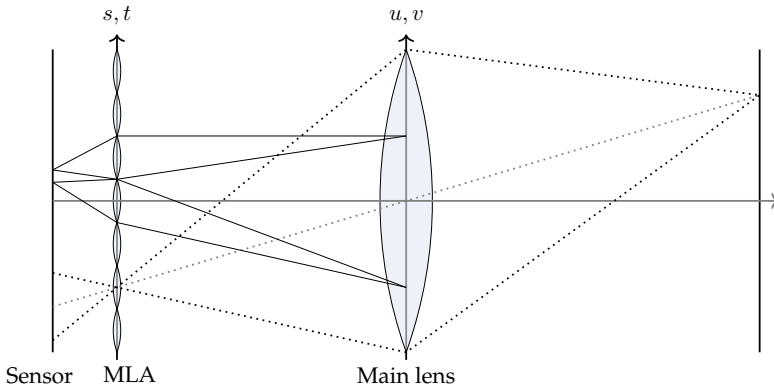


Figure 3.4 Schematic representation of an unfocused plenoptic camera.

100 μm , the main lens is almost infinitely far away when compared to the distance between the sensor and MLA. The rays entering the microlenses can therefore be assumed to be parallel. As a consequence, rays that are imaged onto the central pixel of the sensor region belonging to a microlens originate from the center of the main lens. And rays away from the edge of the main lens are projected onto pixels corresponding to angular coordinates away from the microlens center. Consequently, each u, v -coordinate samples only a sub-area of the camera's aperture. Hence, each SAI shows a very high depth of field, due to the small opening. To avoid overlapping between different microlens images, the f -numbers of the main lens and the microlenses must be matched to each other [137]. In the unfocused design, the spatial resolution is defined by the number of microlenses in front of the sensor, whereas the angular resolution is defined by the number of pixels under each microlens.

Since the sensor's resolution is fixed, the spatial resolution of the light field decreases with increasing angular resolution. Because of this trade-off, new camera designs were introduced that allow a light field to be captured with significantly higher spatial resolution than the traditional approach, enabling the rendering of high-resolution images that meet the expectations of modern photographers [122]. MLA-based light field cameras in the so-called *focused design* were first introduced by Lumsdaine and Georgiev [123], and then later commercialized by Raytrix GmbH. In

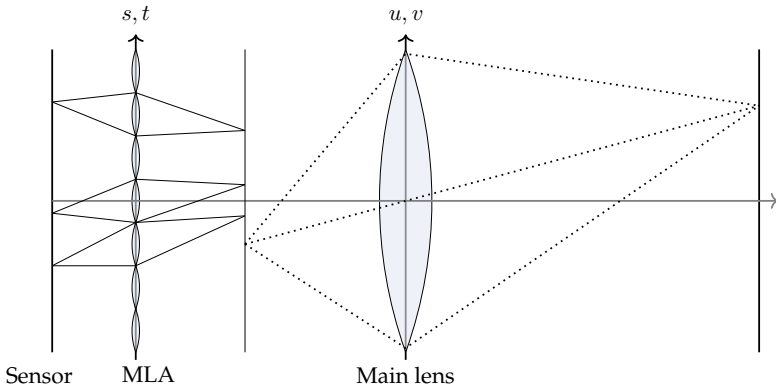


Figure 3.5 Schematic representation of a focused plenoptic camera.

this design, the distance between the MLA and the sensor differs from the focal length of the microlenses, see figure 3.5. Hence, the microlenses don't sample the main lens' aperture but a virtual image plane. The relation between light field coordinates and the optical components of the camera is no longer as intuitive as it was before. With the focused design, the number of pixels under each microlens no longer corresponds directly to the angular resolution. Rather, the microlenses now show micro-images of the scene. Each microlens can therefore be interpreted as a tiny virtual camera, where depending on the position of the microlens, both the optical center of the virtual camera is shifted and a different small section of the scene is observed. The pixels underneath the microlens thus encode spatial information, while the microlens position contains both spatial and angular information, due to a slightly different view of the scene in each micro-image. In particular, there are even different configurations for focused plenoptic cameras. Because the micro-images only scan the virtual image plane onto which the main lens images the scene, the micro-images have a significantly reduced depth of field compared to the unfocused design. This led to the introduction of multi-focus plenoptic cameras, which have micro-lenses with different focus lengths [148]. In this way, a focused image can be constructed at any depth of focus, and a really wide range of digital refocusing can be achieved [61].

Apart from microlens-based light field cameras, a variety of more exotic designs exists, *e.g.*, cameras based on coded apertures [211], multi-spectral light field cameras [178, 179], or light field objectives that can turn any standard camera into a light field system by using kaleidoscope-like imaging optics [128].

3.2.2 Related Works

While a conventional camera only captures the spatial information of a scene, a light field with the additional angular information can be used, for example, to change the perspective on the scene or to change the position of the observer [137]. Moreover, even after capturing a dynamically active scene, it becomes possible to shift the focus plane by rendering new images from the 4D light field data [135]. Due to the highly redundant information, light fields can be used for a variety of applications, *e.g.*, denoising and deblurring [4, 43], super-resolution [19], segmentation [219], material recognition [213], hyper-spectral imaging [181], structure from motion and visual odometry [94, 238], to name a few.

A popular research area is light field-based disparity estimation, which can be used directly to estimate depth if the camera is calibrated. A variety of methods exist for this purpose [98]. Due to the multi-view property of the light field, well-established feature-based approaches comparable to stereo imaging can be used [77]. Approaches for depth from focus/defocus [199] or disparity from EPIs exist [212, 218]. Using the EPIs, lines of constant intensity appear, and disparity estimation can be performed with a local orientation estimate [216] or by a local line fitting [244, 245]. More generally, the disparity information is represented by the two-dimensional slope of constant-intensity planes embedded in the 4D space. In recent years, deep learning approaches have become state of the art in disparity estimation, as they can provide more robust local slope estimation [76, 187], or even incorporate the full 4D light field information into the process [78, 124, 234].

In the field of partially specular reflection (or partial transparency), the EPIs show a superposition of lines representing the direct depth of the partially specular surface and the indirect depth of the reflected scene, respectively. For these situations, many approaches exist to model and remove specular highlights [41], to estimate both depths simultane-

ously [95, 232], or to separate both image layers to obtain two separate light fields [95, 96, 193]. Light fields are also used to detect and classify non-Lambertian objects, such as refractive or transparent objects [126, 224, 225]. Ideguchi *et al.* [91] estimated the surface of transparent objects based on local photo consistency. Lu *et al.* [121] used light fields to sample surface BRDFs and developed an architecture based on convolutional neural networks (CNN) for BRDF identification. Alperovich *et al.* [7] used a deep encoder-decoder network that solves non-Lambertian intrinsic light field decomposition, which can recover albedo, shading, and specularities. Light field cameras have found applications in the field of optical metrology as well. Ziwei *et al.* [252] use a light field camera as an additional geometric constraint to resolve the ambiguity of phase unwrapping, which serves as the basis for many optical metrology applications. Liu *et al.* [119] achieve high dynamic range 3D imaging by using a light field camera for multi-view fringe projection profilometry, Zhou *et al.* [251] combine the light field's EPI-based depth estimation to improve profilometric reconstruction, and Farber *et al.* [58] demonstrate that by using spectral light fields, an application like depth estimation or profilometry can be improved even more.

4 Deflectometric Registration

Deflectometry is used for high-precision surface measurement and dense 3D reconstruction of specular objects. In this context, it is necessary to carry out an optical position encoding to be able to reconstruct the surface by means of triangulation. As described in Sec. 3.1, the objective of the deflectometric registration is to determine an imaging function, which allows direct mapping of camera pixels to points in the monitor plane. With the help of this registration, local defects in the surface under test can be detected or the surface can be reconstructed globally, see Ch. 7. Apart from deflectometry, optical encoding techniques can also be used in the field of camera calibration, where reference features displayed by an active target drastically decrease the calibration error as compared to when standard checkerboard features are used, see Ch. 5. Hence, to ensure precise measurements, the registration must be as accurate as possible. In order to determine the imaging function, the positions on the reference plane and thus the pixel coordinates of the monitor screen must be uniquely assigned to pixels in the camera employing an encoding process.

There are a number of possibilities for such an encoding. In principle, it would be most straightforward to turn on each individual reference pixel one at a time and check which camera pixel is measuring an increase in intensity. However, this would take a considerable amount of time. It makes more sense to encode all reference pixels simultaneously using more advanced methods. A local encoding of the reference pixels can be done by displaying statistical patterns where each position within this pattern is identified by the local pixel neighborhood [173]. While this method enables very fast measurements, since only one pattern has to be displayed, it is only of limited use for the measurement of more complex scenes. Because the surface typically distorts the reference pattern, the encoding of the local neighborhood can often no longer be recognized. To achieve a high-accuracy measurement, a temporal encoding of each pixel is more suitable. Here, instead of a single pattern, a sequence of

patterns is now displayed by the reference. The sequence of intensity values measured in the camera subsequently allows decoding the reference pixels and yields the determination of the imaging function. A popular temporal coding method is the coding of the reference pixels by means of a gray code [159]. Here, a binary pattern sequence is displayed by the reference to uniquely code the individual pixels. However, a major disadvantage of the gray-code method is that it uses only binary intensity values. As a result, the displayed signal with its sharp edges has high-frequency components. Because most of the time the camera and the surface provide a blurred image of the reference pattern, these edges become blurred and the decoding becomes more difficult. Another disadvantage is that only discrete pixels can be encoded and no subpixel information can be extracted [159].

Because of these disadvantages, phase-shift coding methods have become widely accepted in structured illumination applications. Here, a sequence of sinusoidal signals is displayed by the reference, whereby the coding of the pixel coordinate is contained in the phase of the sinusoidal signal. The great advantage of these methods is that they are robust to a variation in the ambient illumination, to noise, to low-pass filtering due to a defocusing effect of the camera, and that they allow an estimation of the phase uncertainty [59]. At the same time, these methods enable a subpixel-accurate encoding if the reference pixels are slightly out of focus. To further increase the accuracy of the measurement, multi-frequency methods are used, where sinusoidal pattern sequences with different frequencies are displayed. While this increases the accuracy of the registration, the periodicity of the sinusoidal pattern sequence leads to an ambiguous position encoding in the entire measurement range with just a single phase measurement. The uniqueness range of the phase measurement initially extends only over one period of the underlying sinusoidal pattern. This leads to a modulo- 2π phase wrapping, which can only be compensated using phase unwrapping methods.

When only one phase measurement is available, spatial unwrapping methods must be used, which examine the local 2D neighborhood of the phase map and use spatial information to unwrap it. For applications where several phase measurements can be performed, the so-called temporal multi-frequency phase unwrapping methods have proven to be

the best choice, since they allow a pixel-individual unwrapping. These temporal methods are generally categorized into four groups: hierarchical methods [88–90, 102, 147, 201], heterodyne methods [37, 42, 107, 150, 158, 169, 170, 203, 204, 214, 253], number-theoretical methods [45, 46, 69, 160, 161, 190, 198, 202, 248], and distance minimization-based methods [54–57, 111, 149, 255]. They differ in the way the unwrapping is performed, in which frequency configurations can be used, and in how large the resulting uniqueness range of the unwrapping is. However, a disadvantage of the classical methods is that typically not all phase measurements are unwrapped at the same time. Moreover, they often do not take into account the inherent periodic structure of the phase, which leads to erroneous results. More importantly, the estimation of the phase uncertainty is completely neglected in the entire unwrapping procedure.

To overcome these deficiencies, this chapter presents a probabilistic approach for phase unwrapping, which uses circular statistics to describe the multi-frequency phase-shift coding to optimally reconstruct the phase. The presented approach respects the periodicity of the phase, implicitly unwraps all phase measurements simultaneously by finding the underlying optimal position encoding that caused the phase measurement using maximum-likelihood estimation, allows for an easy frequency selection with a maximum uniqueness range of the unwrapping, and additionally, includes the estimation of the phase uncertainty into the overall unwrapping process. Furthermore, in this chapter, it is proposed to not only perform a temporal unwrapping but to additionally incorporate the information of the local pixel neighborhood in the modeling and thus obtain a probabilistic approach for spatio-temporal phase unwrapping.

The structure of this chapter is as follows: Sec. 4.1 discusses the general concept of phase-shift coding and shows how the phase and the phase uncertainty can be reconstructed from the sinusoidal pattern sequence. Sec. 4.2 introduces the principles of phase unwrapping. Sec. 4.3 describes how the state-of-the-art phase unwrapping algorithms can be optimized by slight modifications. Eventually, Sec. 4.4 presents the probabilistic approach for phase unwrapping. Finally, in Sec. 4.5 the presented methods are extensively analyzed and compared to the state of the art.

4.1 Phase-Shift Coding

In principle, to obtain an absolute coordinate, one could display a single sinusoidal signal or a linearly increasing intensity curve on the reference system and then assign an intensity value to each pixel. However, since commercially available monitor screens or projectors can only display a limited number of discrete intensity levels (usually only 8 bits), one would have to expect strong quantization errors. Furthermore, this simple approach would be very vulnerable to external influences, such as a variation of the ambient illumination or attenuation of the signal's amplitude. Therefore, it makes more sense not to use the signal intensity as an information carrier but rather the phase of a sinusoidal signal.

The basic principle of phase-shift coding is to assign an individual phase $\varphi_x(x, y)$, $\varphi_y(x, y)$ of sinusoidal signals to each reference pixel (x, y) :

$$x = \frac{\varphi_x(x, y)}{2\pi f}, \quad y = \frac{\varphi_y(x, y)}{2\pi f}, \quad (4.1)$$

where the coordinates of the pixels are interpreted as relative coordinates $x, y \in [0, S)$ with $S = 1$ for the rest of this chapter.

Phase-shift coding must be performed independently in both the horizontal and vertical direction, which is why only the encoding in the x direction is considered in the following. The encoding in the y direction is done analogously. Further, the argument of the phase is also simplified by omitting the coordinate y , since the phase in x direction will take the same value for each y . In other words, in the following $\varphi(x) := \varphi_x(x, y)$ holds without loss of generality.

To encode a normalized monitor coordinate $x \in [0, 1)$, a signal sequence of M sinusoidal patterns with frequency f and shifted by Ψ_m is generated and displayed on a monitor screen, whereby the coordinate is contained in the phase $\varphi(x) = 2\pi f x$ of the signal sequence

$$I_m(x) = \frac{I_{\max}}{2}(1 + \cos(\varphi(x) + \Psi_m)). \quad (4.2)$$

Here I_{\max} represents the maximum displayable brightness value. The type of phase-shift coding is determined by the choice of the discrete phase-shift Ψ_m and can be influenced by the number and also the values of the shifts, see [99, 194] for a comparison of possible methods. In

this work, only the most widely used class of phase-shift algorithms is considered, the so-called symmetric M -step algorithms with equidistant phase offsets

$$\Psi_m = \frac{2\pi m}{M}, m \in [1, 2, \dots, M]. \quad (4.3)$$

The signal I_m displayed on the reference illuminates the scene that is to be examined and is then mapped onto the camera sensor. In the case of deflectometry, the signal is emitted by a monitor screen, reflected at a specular surface, and projected into the camera. When using phase-shift coding to obtain reference features for camera calibration, the camera may directly observe the monitor screen. Thus, regardless of the application, a camera records a signal sequence for every camera pixel $\mathbf{u} = (s, t)^T$

$$\tilde{I}_m(\mathbf{u}) = A(\mathbf{u}) + B(\mathbf{u}) \cos(\varphi(\mathbf{u}) + \Psi_m), \quad (4.4)$$

with $m = 1, \dots, M$. Here $A(\mathbf{u})$ is a constant background illumination, $B(\mathbf{u})$ is the modulation of the signal and $\varphi(\mathbf{u})$ is the phase that contains the information about the encoded screen pixels $x(\mathbf{u})$.

Because each camera pixel \mathbf{u} can be considered independently, the coordinates \mathbf{u} are neglected in the following for clarity.

To determine the three unknown quantities A, B, φ from the recorded signal sequence, at least $M \geq 3$ phase shifts are needed and the formulas for the solutions can then be derived [194]

$$A = \frac{1}{M} \sum_{m=1}^M \tilde{I}_m, \quad (4.5)$$

$$B = \frac{2}{M} \sqrt{\left(\sum_{m=1}^M \tilde{I}_m \sin(\Psi_m) \right)^2 + \left(\sum_{m=1}^M \tilde{I}_m \cos(\Psi_m) \right)^2}, \quad (4.6)$$

$$\varphi = \arctan2 \left(- \sum_{m=1}^M \tilde{I}_m \sin(\Psi_m), \sum_{m=1}^M \tilde{I}_m \cos(\Psi_m) \right), \quad (4.7)$$

where $\arctan2(a, b) \in [-\pi, \pi)$ is used, which correctly assigns the arguments of the arctangent to the four quadrants. Also, for sake of simplicity, in the remainder of this chapter the domain of the phase is shifted to positive values:

$$\varphi \equiv \varphi \bmod 2\pi \in [0, 2\pi). \quad (4.8)$$

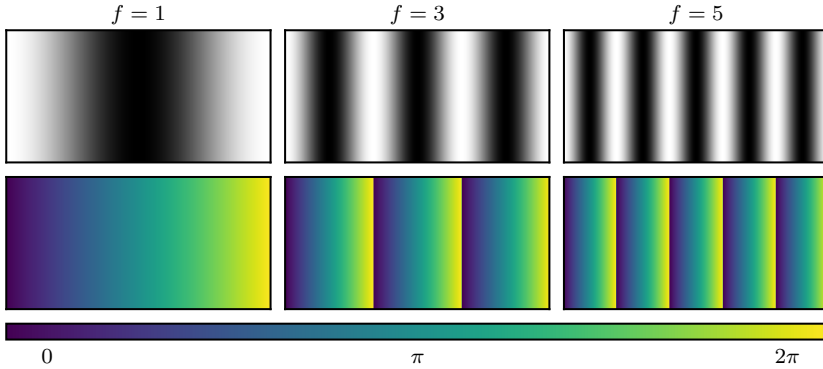


Figure 4.1 Top: Displayed cosine pattern with $\Psi = 0$. Bottom: Corresponding phase maps. The phase is wrapped for $f > 1$.

From equations (4.5), (4.6), (4.7) it becomes clear that the encoding of the phase is robust to many external influences. A locally variable ambient illumination would only affect the offset A . Attenuation of the signal amplitude by, for example, a dark surface would only reduce the contrast of the signal, resulting in a smaller modulation B . The important information, the phase φ , however, remains in principle completely unaffected by this. Furthermore, since the sinusoidal pattern sequence consists only of single signal components with the frequency f but has no higher frequency components, the method is also very robust against low-pass filtering caused by blurring. It can be shown that only the modulation B is reduced, whereas the phase φ remains unaffected. To be more precise, it is even advantageous to slightly image the pattern sequence out of focus, as this blurs the individual pixels of the reference pattern and allows subpixel accuracy to be achieved in the encoding [182].

4.1.1 Phase Uncertainty

The accuracy of the phase measurement is influenced by external systematic influences of the entire measurement setup as well as by stochastic errors. For example, the nonlinearity of the intensity characteristic of the reference system can degrade the phase measurement. This however can be easily compensated using gamma calibration procedures or by using

phase-shift coding with more shifts [117, 241]. Therefore, it will not be a subject of further consideration in this work. Other external systematic influences may change the brightness and contrast of the pattern sequence, which can lead to an increase in uncertainty. For example, the camera optics can image the sinusoidal patterns out of focus, which leads to a decrease in contrast. As the surface is usually part of the structured illumination system, the shape, roughness, and color of the surface also influence the quality of the estimation. Due to these system-related influences, the uncertainty of the phase estimation can be different for each pixel. Furthermore, the phase measurement is influenced by stochastic errors. Every camera image is accompanied by image noise. It is obvious that this noise also affects the phase estimation and influences the uncertainty of the measurement. In general, the sensor noise shows up as noise in the pixel values and can be regarded in a good approximation as normally distributed noise with variance σ_I^2 and zero mean [226].

Li *et al.* [112] show that the phase noise ε_φ can be calculated through Gaussian error propagation from the noise ε_{I_m} of the images of the pattern series:

$$\varepsilon_\varphi = \sum_{m=1}^M \partial_{\tilde{I}_m} \varphi|_{\varepsilon_{I_m}=0} \varepsilon_{I_m} = \sum_{m=1}^M \frac{2 \sin(\varphi + \Psi_m)}{BM} \varepsilon_{I_m}. \quad (4.9)$$

Further, for symmetrical M -step methods, the phase noise has zero mean and its uncertainty, *i.e.*, the standard deviation of ε_φ , can be specified:

$$\sigma_\varphi = \frac{2}{M} \frac{\sigma_I}{B} \sqrt{\sum_{m=1}^M \sin^2\left(\varphi + \frac{2\pi m}{M}\right)} = \sqrt{\frac{2}{M}} \frac{\sigma_I}{B}. \quad (4.10)$$

While B and M can be estimated or are directly defined by the phase-shift coding, the sensor noise is initially unknown. To be able to describe the phase noise absolutely, Fischer *et al.* [59] introduced a quantitative noise model, which combines the phase noise with the parameters of the EMVA 1288 standard for camera systems [226]. This makes it possible to predict the phase uncertainty very precisely by calculating only the modulation B from the pattern sequence.

To further reduce the uncertainty, it is useful to use sinusoidal pattern sequences with a frequency $f > 1$. This has two beneficial effects. The

first effect is a reduction in the quantization noise. Because conventional monitors or projectors can be operated with 256 intensity levels, sinusoidal patterns with small frequencies show considerable steps in the signal (*e.g.*, 1920 pixels of a monitor screen cannot be described without ambiguities when only 256 intensity values are used). Increasing the frequency provides locally a higher dynamic in the pattern. This attenuates the influence of quantization errors [59]. The second and more important effect is a reduction of the phase noise induced by the camera sensor noise. As explained in more detail in the next section, phase jumps occur in the reconstructed phase when the frequency of the sinusoidal pattern sequence is chosen to be $f > 1$. The phase would take values $\varphi > 2\pi$ but is only defined on the periodic interval $[0, 2\pi)$. Thus, the real line \mathbb{R} is wrapped to the smaller interval $[0, 2\pi)$, see figure 4.1. To unwrap the phase again, an integer multiple of 2π must be added at corresponding places, see Sec. 4.2. The unwrapped phase finally results in

$$\Phi_f = \varphi + 2\pi k + \varepsilon_\varphi, \quad (4.11)$$

where $\varphi \in [0, 2\pi)$ represents the wrapped phase, $k \in \mathbb{Z}$ is the unwrapping factor and $\varepsilon_\varphi \in [0, 2\pi)$ represents the phase noise with uncertainty σ_φ . Since the domain of the unwrapped phase has been increased to $\Phi_f \in [0, 2\pi f)$, it has to be scaled back to the original range. The final phase measurement therefore results in

$$\Phi = \frac{\Phi_f}{f} = \frac{\varphi + 2\pi k}{f} + \frac{\varepsilon_\varphi}{f}, \quad (4.12)$$

with $\Phi \in [0, 2\pi)$. By increasing the frequency and then scaling back, the phase information is not changed, but the noise is reduced by the factor $1/f$. The uncertainty of the unwrapped phase is then be given by

$$\sigma_{\varphi,f} = \frac{1}{f}\sigma_\varphi = \frac{1}{f}\sqrt{\frac{2}{M}}\frac{\sigma_I}{B}. \quad (4.13)$$

In summary, with the phase-shift coding one obtains not only a pure position encoding but additionally also the associated uncertainty, where the complete information is encoded in the phase φ and the phase uncertainty σ_φ .

4.2 Principles of Phase Unwrapping

If the frequency of the phase-shift pattern sequence is chosen to $f > 1$, jumps occur in the reconstructed phase. These jumps appear whenever the phase would exceed the value 2π but is mapped back to the interval $[0, 2\pi)$ by the arctangent (4.7). Resolving these jumps is the goal of phase unwrapping. For this purpose, an integer multiple k of 2π is added to the wrapped phase, which is often called the period-order number or unwrapping factor. Since the wrapping of the phase strongly depends on the chosen frequency, the optimal choice of the unwrapping factor is also frequency-dependent. For a coordinate x and frequency f_i , (4.11) can be rewritten:

$$\Phi_i(x) = \varphi_i(x) + 2\pi k_i(x), k_i \in [0, 1, \dots, \lceil f_i \rceil - 1] . \quad (4.14)$$

The task of phase unwrapping is to find the correct k_i for each phase measurement. Since an individual unwrapping factor exists for each pixel, the problem from (4.14) is initially under-determined. To get a solution anyway, additional information has to be used. In principle, there are two approaches to solve the problem: spatial and temporal phase unwrapping.

Spatial phase unwrapping algorithms are useful when it cannot be guaranteed that the phase remains constant over time or when repeated measurements would be too costly. With spatial algorithms, phase unwrapping is performed using only a single phase measurement. The information necessary for the unwrapping is then obtained from the 2D pixel neighborhood. For example, in region growing-based approaches, starting from an initial pixel, the phase is unwrapped aiming to achieve a continuous phase profile where neighboring pixels have a similar value [183, 243, 250]. However, spatial unwrapping is very susceptible to noise, and phase discontinuities can make the unwrapping difficult or cause errors. For example, a step in the phase cannot be reconstructed without ambiguity, since the algorithm is unable to determine the step's height, which may have a multiple of 2π as an offset. The main disadvantage of spatial unwrapping methods is that they can generally only obtain a relative phase instead of an absolute one, which is not useful for 3D reconstruction problems. Hence, if the requirements for spatial phase

unwrapping are not satisfied or an absolute phase estimate is needed, temporal phase unwrapping must be used.

While this work is focused on phase-shift coding, these phase wrapping effects also appear in other fields of optical metrology, *e.g.*, interferometry [37, 42, 158], SAR imaging [40, 155], or even time-of-flight imaging [47, 48]. Thus, the phase unwrapping problem influences many other applications.

4.2.1 Temporal Phase Unwrapping

Temporal phase unwrapping methods in general do not use the spatial information in the phase map. They can therefore handle each pixel individually, which means that discontinuities in the phase do not cause any problems. On the other hand, they rely on additional information obtained by additional measurements. This can be achieved, for example, by recording additional image patterns that can be decoded unambiguously. Methods based on temporal gray-coding achieve unambiguous coding and can be used as a basis for phase unwrapping [175, 240]. However, they cannot achieve sub-pixel accuracy and are susceptible to noise and defocusing effects [254]. An encoding using statistical patterns allows spatial decoding, which can be used directly for phase unwrapping. While these methods allow for a fast acquisition time, the evaluation of statistical patterns has similar drawbacks as the spatial phase unwrapping algorithms. For an overview of absolute phase unwrapping methods, the reader is referred to the literature [242, 254].

This work is focused on another class of unwrapping methods: Temporal multi-frequency phase unwrapping. These methods use multiple phase-shift pattern sequences with different frequencies f_i to obtain multiple phase measurements $\varphi_i \in [0, 2\pi)$, all of which are based on the same coordinate encoding. Depending on the frequencies, the phase measurements are wrapped differently. Since it is assumed that the unwrapped phase does not change over time, the multiple phase measurements generate a system of equations, where each equation has the form of (4.14).

Because all phase measurements are based on the same coordinate x , if certain requirements are met, the equation system has a unique solution

$$x \equiv \frac{\Phi_i}{2\pi f_i} = \frac{\varphi_i}{2\pi f_i} + \frac{k_i}{f_i}. \quad (4.15)$$

The unwrapping of the phase measurements is then obtained by solving this equation system, for which various methods exist.

4.2.1.1 Hierarchical Unwrapping

The hierarchical methods are among the most intuitive approaches. They use a series of phase measurements in which the frequency of the underlying sinusoidal signals is increased in each step. To obtain an unambiguous unwrapping of all phase measurements, the frequency of the first measurement is chosen in such a way that the measured phase is not subject to ambiguities. Thus, $f_0 = 1$ and $\Phi_0 = \varphi_0$. Each subsequent measurement is then unwrapped using the previous unwrapped phase associated with the lower frequency as a reference $\Phi_{\text{ref}} = \Phi_{i-1}$, $f_{\text{ref}} = f_{i-1}$. The unwrapping factor can hereby be determined using a simple rounding operation

$$k_i = \left\lceil \frac{\frac{f_i}{f_{\text{ref}}} \Phi_{\text{ref}} - \varphi_i}{2\pi} \right\rceil, \quad (4.16)$$

and the respective phase is unwrapped with

$$\Phi_i = \varphi_i + 2\pi k_i. \quad (4.17)$$

There are many variations of hierarchical unwrapping algorithms in the literature, which differ mainly in the choice of the frequency sequence, *e.g.*, linearly increasing frequencies [88, 90], exponentially increasing frequencies [89, 147], reversed sequences [90, 102] or generalized approaches [201]. Usually, after unwrapping the individual phase maps, the phase corresponding to the highest frequency is used or all phase maps are averaged.

4.2.1.2 Heterodyne Unwrapping

The two-wavelength heterodyne methods were originally developed for interferometry [37, 42, 158] but are also applicable to phase-shifting 3D-measurement systems [169, 170, 253]. Unlike before, the heterodyne method can be implemented directly for high frequencies. Usually, only two frequencies f_1 and f_2 are used. The phase measurements associated with the two frequencies are subtracted

$$\varphi_{12} = \varphi_1 - \varphi_2 \mod 2\pi, \quad (4.18)$$

and the frequency of the synthetic phase φ_{12} is then given by

$$f_{12} = |f_1 - f_2|, \quad (4.19)$$

where f_{12} represents the beat frequency. If f_1 and f_2 are well-chosen, the uniqueness range of the phase unwrapping can be increased enough to resolve the ambiguity [204]. With the normalized reference size $S = 1$ that is used in this chapter, it can be shown that $f_{12} = |f_1 - f_2| \leq 1$ must hold in order to allow an unambiguous phase reconstruction.

Since the phase noise of φ_1 and φ_2 is accumulated during the formation of the synthetic phase, the signal-to-noise ratio deteriorates. For this reason, the synthetic phase is generally used only to unwrap the underlying measurements φ_1 and φ_2 . The unwrapping factors k_1 and k_2 are hereby calculated using (4.16) with $f_{\text{ref}} = f_{12}$, $\Phi_{\text{ref}} = \varphi_{12}$.

The extension to more than two frequencies is described in [107, 214] and allows increasing the unambiguous measurement range even further. For this, several approaches exist that optimize the choice of the frequencies to obtain a robust unwrapping result [150, 203, 204].

4.2.1.3 Number-Theoretical Unwrapping

The number-theoretical unwrapping methods are based on number theory, relative primes, and the divisibility properties of integers. They were originally proposed by Gushov and Solodkin [69]. They were then further improved to reduce the susceptibility to phase errors [160, 198, 202, 249]. In its basic form, the method uses the Chinese-remainder theorem to

calculate a simultaneous solution to the unwrapping problem. Following the theorem, a system of simultaneous equations of congruence

$$X \equiv b_i \pmod{m_i}, \text{ for } i = 1, \dots, n \quad (4.20)$$

has a unique solution $X \in \mathbb{Z}$, if $b_i \in \mathbb{Z}$ and $m_i \in \mathbb{Z}$ are known integers, where the set of m_i are pairwise co-prime numbers, *i.e.*, for their greatest common divisor applies $\gcd(m_i, m_j) = 1, \forall i, j$. The solution itself is then given by

$$X \equiv \sum_i M_i M'_i b_i \pmod{m}, \quad (4.21)$$

where $m, M_i, M'_i \in \mathbb{Z}$ with

$$m = \prod_i m_i, M_i = \frac{m}{m_i}, M_i M'_i \equiv 1 \pmod{m_i}, \quad (4.22)$$

and where the M'_i can be found using, *e.g.*, the extended Euclidean algorithm [39]. The theorem can be applied to the phase unwrapping problem, by comparing (4.15) to (4.20) and substituting

$$X := xS \equiv \frac{\Phi_i S}{2\pi f_i}, b_i := \left\lceil \frac{\varphi_i S}{2\pi f_i} \right\rceil, m_i := \frac{S}{f_i}. \quad (4.23)$$

If the condition $\text{lcm}(m_1, m_2, \dots) \geq S$ for the least common multiple is fulfilled, the phase ambiguity can then be resolved [115]. Hereby, an appropriate scaling factor S needs to be chosen to obtain meaningful integer values and co-primes m_i . In the case of a deflectometry application, it can be set to the size of the monitor screen measured in pixels. Further improvements to the algorithm can be achieved by precalculating a look-up table to speed up the computation time [45, 46, 161, 190, 248].

4.2.1.4 Distance Minimization-Based Unwrapping

The previous methods have relatively high restrictions on the choice of frequencies. Thus, newer approaches try to circumvent these restrictions by posing the phase unwrapping as an optimization problem. Pribanić *et al.* [161] extend the two-wavelengths number-theoretical method by removing the restriction of having co-prime wavelengths. From the combination of all possible unwrapping factors, they search for the one that minimizes the distance between the two respective unwrapped phases.

The excess fraction methods can be regarded as a multi-wavelength extension of the heterodyne methods [54–57]. They define an excess fraction as the difference between an ideal continuous unwrapping factor and its integer analogon. The unwrapping factors are then determined individually by minimizing the respective excess fraction, where each excess fraction is influenced by all phase measurements.

More recent approaches try to perform the unwrapping of all phase measurements simultaneously to find an ideal solution for all unwrapping factors at the same time. For this purpose, the vector of ideal unwrapping factors $\mathbf{k} = (k_1, k_2, \dots)$ is sought that minimizes the distance of the individual unwrapped phases to the mean value of all unwrapped phase measurements. Here, the distance measure can be defined by an orthogonal projection of the wrapped phases onto a subspace [149], or it can be written down directly as a sum of distances between the unwrapped phases to the averaged unwrapped phase [111, 255]. It is hence titled projection distance minimization (PDM).

With $\Phi = (\Phi_1, \Phi_2, \dots)^T$, $\Phi_i = \varphi_i + 2\pi k_i$, $\mathbf{f} = (f_1, f_2, \dots)^T$ and by minimizing the projection distance

$$\mathbf{k} = \arg \min_{\mathbf{k}} \|\Phi - \mathbf{P}\Phi\|^2, \text{ with } \mathbf{P} = \frac{\mathbf{f}\mathbf{f}^T}{\|\mathbf{f}\|^2}, \quad (4.24)$$

the unwrapping factors, and thus, the simultaneous unwrapping of all phase measurements can be obtained. Here, $\mathbf{P}\Phi$ represents the projection of unwrapped phase measurements, which for the ideal choice of \mathbf{k} should be equal to Φ . The optimal unwrapping factors are thereby found by an excessive trial and error of all possible combinations. To speed up the optimization, Petković *et al.* [149] suggest ignoring impossible combinations and Zuo *et al.* [255] use the geometry of the measurement setup of a profilometry system to further exclude unreasonable combinations.

4.3 Improving the Phase Unwrapping Algorithms

The classical phase unwrapping algorithms from the previous sections do not use all of the information to unwrap the phase measurements. Far more importantly, they generally do not take into account the inherently periodic structure of the phase, which can lead to incorrect unwrapping.

For example, the simple hierarchical unwrapping method from the last section only uses the previous phase measurement with a lower frequency to unwrap the current phase. However, phase measurements with higher frequency could also contain information to unwrap the phase measurements with lower frequency. In addition, the periodic structure of the phase is not taken into account, so unwrapping errors often occur near the 2π -discontinuities. To achieve good accuracy for 3D reconstruction, phase maps corresponding to high frequencies are needed. But then the number of necessary measurements is high because the sequence always starts at $f = 1$. The heterodyne method does not have to start at low frequencies but can directly select high ones, achieving an overall smaller mean uncertainty with the same number of measurements [150]. However, it is disadvantageous that the unambiguous measurement range of the unwrapping is determined by the beat frequency. Thus, there are frequency configurations that do not yield an unambiguous solution but could be solved unambiguously with other methods [149]. Additionally, the method is not straightforwardly extendable to a multi-frequency approach. The number-theoretical unwrapping methods perform a simultaneous unwrapping of all phase measurements and also consider the periodicity of the phase. Nevertheless, the restriction to pairwise co-prime wavelengths makes the selection more difficult, and due to the integer arithmetic and rounding operations, these methods are relatively susceptible to noise [198]. Even more, for the method to work, the frequencies must be chosen very precisely proportional to the integer co-prime wavelengths, which is especially problematic for applications where the wavelengths cannot be chosen freely, *e.g.*, interferometry [53]. The PDM method, on the other hand, performs a simultaneous unwrapping of all phase measurements without having to apply rounding operations. In its current form, however, it is still not perfect. It does not take into account the periodic structure of the phase so that unwrapping errors occur frequently near the boundaries of the coding interval. Also, it is a very expensive procedure due to testing all possible combinations of unwrapping factors k_i . Additionally, all methods have in common that the phase unwrapping does not consider the estimated phase uncertainty at all, although it could help to compensate for an unfavorable measurement. Therefore, the following sections

present improvements to the classical phase unwrapping algorithms to deal with their shortcomings.

4.3.1 Weighted Circular Mean

Multi-frequency phase unwrapping gives different estimates of the phase Φ_i with different uncertainty $\sigma_{\varphi_i, f_i} = \frac{1}{f_i} \sigma_{\varphi_i}$. In many practical applications, only the phase with the highest frequency is used, or in the better case, a weighted average of all phase measurements is calculated. Here either the frequency is chosen as the weighting factor or the inverse of the estimated uncertainty (4.13) is used. Usually, in the sense of an unbiased estimator, the variance of the measurement is used as weighting to obtain the phase

$$\Phi = \frac{\sum_i \sigma_{\varphi_i, f_i}^{-2} \Phi_i}{\sum_i \sigma_{\varphi_i, f_i}^{-2}}. \quad (4.25)$$

However, this very common approach ignores an important property of the phase. The phase is periodic on the interval $[0, 2\pi)$. Since the phase measurement is affected by noise, a true phase value of $\varphi = 0$ can, for example, be estimated as the value $\varphi_1 = 0.01$ in one measurement and as $\varphi_1 = 1.99\pi$ in a second measurement. As a result, the mean value is not $\varphi \approx 0$ as expected but $\varphi = \frac{1}{2}(\varphi_1 + \varphi_2) \approx \pi$. Therefore, very large errors appear at the boundary of the coding interval. A commonly used workaround for this problem is to artificially reduce the used encoding interval. That means instead of displaying phase values in the range $[0, 2\pi)$ on the screen, only the values $[\Delta_\varphi, 2\pi - \Delta_\varphi)$ are used. Depending on the expected noise, an optimal size can even be determined for $\Delta_\varphi \in (0, \pi)$ [149, 150]. However, a reduction of the used interval leads to a lower SNR in the remaining part, due to the effective frequency of the represented sinusoidal signal being reduced.

With this in mind and since the phase is periodic in the interval $[0, 2\pi)$, the usual arithmetic mean must not be used. Instead, this work proposes to use a circular mean value \mathcal{M}° , which is formed by mapping the phase

measurements to the complex unit circle $z_i = e^{j\varphi_i}$ and by calculating a weighted mean of the complex pointers

$$z = \frac{\sum_i \sigma_{\varphi_i, f_i}^{-2} z_i}{\sum_i \sigma_{\varphi_i, f_i}^{-2}}, \quad (4.26)$$

where the circular mean of the phase can then be calculated from the argument of the resulting complex pointer

$$\begin{aligned} \mathcal{M}^\circ(\Phi, \sigma_{\varphi, f}) &:= \arctan2(\text{Im}(z), \text{Re}(z)), \\ &= \arctan2\left(\sum_i \frac{\sin(\Phi_i)}{\sigma_{\varphi_i, f_i}^2}, \sum_i \frac{\cos(\Phi_i)}{\sigma_{\varphi_i, f_i}^2}\right). \end{aligned} \quad (4.27)$$

Additionally, the uncertainty of the mean phase

$$\sigma_\varphi^2 = \frac{1}{\sum_i \frac{1}{\sigma_{\varphi_i, f_i}^2}} \quad (4.28)$$

can be estimated to be further used in any subsequent application.

4.3.2 Modified Hierarchical Unwrapping

A disadvantage of the standard hierarchical phase unwrapping is that the phase measurements belonging to higher frequencies are unwrapped solely with the help of the previously unwrapped phase measurement. For the case of more than two used frequencies, it makes sense to modify the standard approach to make the unwrapping more robust against errors. It is advisable to use not only the last phase measurement as a reference, but the average of all phases already processed. The more frequencies are used, the more the method will benefit from all the previous unwrapped phases. For the averaging operator, the weighted circular mean from the previous section is used. With it, the periodicity of the phase can be partially compensated, and by using the phase uncertainty as a weighting factor, the overall unwrapping is improved due to penalizing low-quality phase measurements. The standard hierarchical algorithm is relatively easy to adjust to obtain the modified hierarchical unwrapping. Algorithm 1 shows the procedure.

Algorithm 1 Modified Hierarchical Unwrapping

Input: Wrapped phase maps φ , frequencies \mathbf{f} (with $f_0 = 1$), phase uncertainties σ_φ

Output: Fusion of unwrapped phase maps Φ

- 1: Set $\Phi_{ref} = \Phi_0 = \varphi_0 \bmod 2\pi$
- 2: **for** $n = 0, 1, \dots, N-1$ **do**
- 3: Get unwrapping factor
- 4: $k_n = \left\lceil \frac{f_n \Phi_{ref} - \varphi_n}{2\pi} \right\rceil$
- 5: Unwrap current phase map
- 6: $\Phi_n = \frac{\varphi_n + 2\pi k_n}{f_n} \bmod 2\pi$
- 7: Calculate new reference with circular mean of previous estimates
- 8: $\Phi_{ref} = \mathcal{M}^\circ \left((\Phi_0, \dots, \Phi_n)^T, (\sigma_{\varphi_0, f_0}, \dots, \sigma_{\varphi_n, f_n})^T \right)$
- 9: **end for**
- 10: Calculate circular mean of all unwrapped phases
- 11: $\Phi = \mathcal{M}^\circ (\Phi, \sigma_{\varphi, \mathbf{f}})$

4.3.3 Modified PDM Unwrapping

The PDM phase unwrapping method from Sec. 4.2.1.4 attempts to unwrap the phase by minimizing the distance between the vector of phase measurements and a projected version of the same. This can be interpreted as minimizing the distance between each unwrapped phase to the averaged unwrapped phase. By rewriting (4.24) it follows

$$\begin{aligned}
 \|\Phi - \mathbf{P}\Phi\|^2 &= \left\| \Phi - \frac{\mathbf{f}\mathbf{f}^T}{\|\mathbf{f}\|^2} \Phi \right\|^2 = \left\| \Phi - \mathbf{f} \frac{\sum_j f_j^2 \left(\frac{\Phi_j}{f_j} \right)}{\sum_j f_j^2} \right\|^2 \\
 &= \sum_i (\Phi_i - f_i \Phi_{\text{Mean}})^2, \tag{4.29}
 \end{aligned}$$

where $\Phi_i = \varphi_i + 2\pi k_i$ and where the frequencies are used as weighting factor in Φ_{Mean} . To improve the method only three simple modifications are necessary. First, the weighting factor is replaced by the squared inverse of the frequency-dependent phase uncertainty σ_{φ_i, f_i} . Further, to respect

the periodicity of the phase, the presented circular mean is used. And at last, instead of using a classical distance measure, a circular distance

$$d^\circ(\Phi_a, \Phi_b) := \pi - |\pi - |\Phi_a - \Phi_b|| \quad (4.30)$$

is used, which returns the smallest distance in a periodic interval between two points $\Phi_a, \Phi_b \in [0, 2\pi)$. Phase unwrapping using this modified PDM method is then achieved by finding the optimal unwrapping factors

$$\mathbf{k} = \arg \min_{\mathbf{k}} \sum_i d^\circ(\Phi_i, \mathcal{M}^\circ(\Phi, \sigma_{\varphi, \mathbf{f}}))^2. \quad (4.31)$$

4.4 Probabilistic Approach for Temporal Phase Unwrapping

The modified hierarchical procedure presented in the previous section attempts to integrate the uncertainty measurement into the unwrapping as a first step by using it in the weighted average calculation, partially respecting the periodicity of the phase by using the proposed circular mean. However, a complete and simultaneous unwrapping of all phase measurements is not given here either. The proposed modified PDM unwrapping respects the periodicity of the phase. However, because all combinations of the unwrapping factors need to be evaluated, it is computationally extremely expensive. Moreover, it is by no means clear whether the minimization of the squared circular distance yields an optimal unwrapping result. Therefore, this work proposes a completely different idea that addresses the phase unwrapping problem through a probabilistic approach.

In the field of phase unwrapping, probabilistic approaches have already been used in the spatial domain. Carballo and Fieguth [32] and Koetter *et al.* [105] use a probabilistic approach to model the probability of a phase discontinuity in interferometric synthetic aperture radar (InSAR) images to use them as weight factors for a spatial phase unwrapping procedure. Droschel *et al.* [48] use a similar approach for time-of-flight imaging. Baselice *et al.* [14] use an extended Kalman filter that includes probabilistic data to perform phase unwrapping and phase noise reduction of InSAR data.

In contrast to these approaches, a probabilistic model for temporal phase unwrapping is proposed here. To solve the phase unwrapping problem optimally, an attempt is made to find the coordinate that has the highest probability of having caused the corresponding phase measurements. To formulate the unwrapping as a probability problem, the phase measurement is modeled as an appropriate stochastic process. This is used to determine the probability density of the encoded coordinate, find the optimal decoding by a maximum-likelihood approach, and thus implicitly and simultaneously compensate for the wrapping of all phase measurements.

4.4.1 Probability Density Function of Phase-Shift Coding

As indicated in Sec. 4.1.1, the variance of the image noise can be propagated through the phase-shifting algorithm. Thus, every measurement provides not only an estimate of the phase φ but also the uncertainty σ_φ of this estimation. The probability density function of the true phase is therefore centered around the respective measurement. The question now arises which probability distribution the phase has. In principle, several distribution functions are possible. Since the image noise has a normal distribution, the first assumption is that the phase noise is also normally distributed. However, because the phase has a periodic structure and is only defined on the interval $[0, 2\pi)$, the probability density must be searched in the field of circular statistics [93].

4.4.1.1 Wrapped Normal Distribution

The most intuitive approach to obtain a probability distribution of the phase is to assume a normal distribution $\theta \sim \mathcal{N}(\mu, \sigma^2)$ and to allow its values to be spread on the entire set of real numbers $\theta \in \mathbb{R}$. By folding the density function around the unit circle

$$\varphi = \theta \bmod 2\pi, \quad (4.32)$$

the range of values is then forced to the interval $[0, 2\pi)$. The density function of the folded random variable is then the wrapped normal distribution [93]

$$p_{\text{WN}}(\varphi) = \sum_{k=-\infty}^{\infty} \mathcal{N}(\mu + 2\pi k, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \sum_{k=-\infty}^{\infty} e^{\frac{-(\varphi - \mu - 2\pi k)^2}{2\sigma^2}}, \quad (4.33)$$

with the parameters $\mu \in [0, 2\pi)$ und σ^2 . The density function is symmetric and centered around the expected value μ , whereas the width of the function is affected by the parameter σ . Since in practice the infinite sum must be terminated at some point, the literature provides more efficient representations of the distribution, *e.g.*,

$$p_{\text{WN}}(\varphi) = \frac{1}{2\pi} \left(1 + 2 \sum_{p=1}^{\infty} e^{\frac{-\sigma^2 p^2}{2}} \cos(p(\varphi - \mu)) \right), \quad (4.34)$$

where, depending on the choice of σ^2 , the sum can be aborted after only a few terms [106].

4.4.1.2 von Mises Distribution

A major disadvantage of the wrapped normal distribution is that it is quite intractable due to the infinite sum. Furthermore, it is not assured that a real phase measurement results from a folding operation on a linear normal distribution around the unit circle. Hence, it is not mandatory to assume that (4.32) is the correct description of the phase-shift coding.

If the problem is approached with minimal knowledge, an alternative probability density function for the phase can be found. The available knowledge is: the expected value of the distribution corresponds to a phase measurement μ , there is a measure of the second central moment σ_φ , and the phase should be defined on the periodic interval $[0, 2\pi)$. The circular probability density function which maximizes the entropy under the given conditions and thus represents the ideal choice under these circumstances is the *von Mises* distribution [93]

$$p_{\text{vM}}(\varphi) = \frac{e^{\kappa \cos(\varphi - \mu)}}{2\pi I_0(\kappa)}, \quad (4.35)$$

where $I_0(\kappa)$ is the modified Bessel function of the first kind and order zero

$$I_0(\kappa) = \frac{1}{2\pi} \int_0^{2\pi} e^{\kappa \cos(\theta)} d\theta = \sum_{r=0}^{\infty} \left(\frac{\kappa}{2}\right)^{2r} \left(\frac{1}{r!}\right)^2. \quad (4.36)$$

The parameter μ represents the expected value and κ depicts a concentration measure that is analogous to the inverse of the variance in the normal distribution. Because of its mathematical simplicity, the *von Mises* distribution is one of the most commonly used distributions in circular statistics. And due to its great importance, it is also often referred to as the circular normal distribution [93].

4.4.1.3 Phase Noise Model of Phase-Shift Coding

A more precise way to describe the probability density of the phase is to analyze the phase-shift coding directly. Rathjen [167] examines the random phase error arising from the normally distributed image noise of the sinusoidal pattern sequence. The two arguments of the $\arctan2$ function from (4.7) are described using a bivariate normal distribution, where the parameters of the distribution are computed from the normal distribution of the image noise of the underlying pattern sequence. Finally, the distribution of the phase is calculated from this bivariate normal distribution, which applies to any phase-shift coding method.

Depending on the algorithm, different distributions are obtained, which do not necessarily have to be symmetrical and which may also depend on the absolute value of the phase. For the symmetric M -step methods used in this work, the arguments of the $\arctan2$ -function are uncorrelated and have the same variance, leading to a symmetric distribution function for the phase that is independent of the absolute phase value [167]. The probability distribution function of the phase φ for symmetric M -step algorithms is then given by

$$p_{\text{PM}}(\varphi) = \frac{e^{-\text{SNR}}}{2\pi} \left\{ 1 + \sqrt{\pi \text{SNR}} \cos(\varphi - \mu) e^{\text{SNR} \cos(\varphi - \mu)^2} \cdot \left(1 + \text{erf} \left(\sqrt{\text{SNR}} \cos(\varphi - \mu) \right) \right) \right\}, \quad (4.37)$$

where $\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$ is the Gaussian error function, the signal-to-noise ratio $\text{SNR} = \frac{1}{2} \sigma_\varphi^{-2}$ determines the width of the distribution, and $\mu \in [0, 2\pi)$ represents the expected value.

4.4.1.4 Comparison

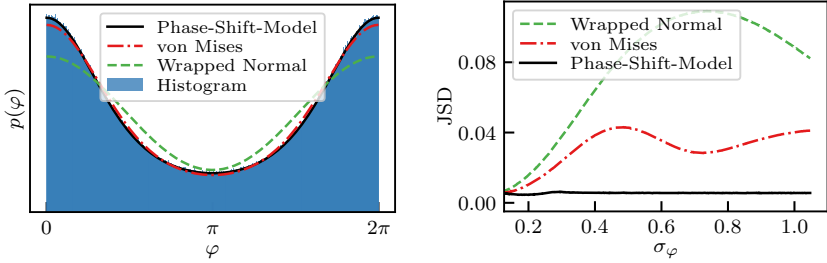
To identify which of the presented distributions is best suited for the problem, a Monte Carlo simulation of the phase measurement is performed. For this, the coordinate $x = 0$ is encoded using phase-shift coding and then the phase φ is measured. The simulation is performed 10^7 times, each time adding Gaussian image noise corresponding to a phase uncertainty of $\sigma_\varphi = \frac{\pi}{2}$. The probability density of the phase noise can then be approximated using histogram analysis.

Figure 4.2(a) shows the histogram of the measured phases and the different density functions whose parameters can be calculated from the phase-shift coding. As expected, the phase noise can best be described by the noise model of Rathjen [167]. However, the *von Mises* distribution also shows a reasonably good fit to the histogram, whereas the wrapped normal distribution is too low on the hills and too high in other areas of the histogram.

Figure 4.2(b) shows the *Jensen-Shannon* distance (JSD) [49] between the histogram and each of the distributions over different phase uncertainty values as a similarity measure, *i.e.*, a small JSD value corresponds to a high similarity. It can be seen that the model of Rathjen has a high similarity to the histogram for all uncertainty values. The *von Mises* distribution is also very close to the histogram and hence, represents the phase-shift coding sufficiently good, although the similarity is not constant for all noise values. Finally, compared to those two distributions, the wrapped normal distribution has a greater distance to the histogram. For small noise values σ_φ , all distributions converge into one another [93], so that they are almost equivalent, and for very large noise values everything converges to the uniform distribution on the interval $[0, 2\pi)$.

4.4.2 Compound Probability Density Function

Because the individual phase measurement is affected by phase noise, the probability density of the true phase is hence centered around the



(a) Histogram of phase noise with $\mu = 0$ and $\sigma_\varphi = \frac{\pi}{2}$ and the three analyzed probability density functions.

(b) Jensen-Shannon distance to the histogram for different values of the phase uncertainty.

Figure 4.2 Comparison of phase noise models.

respective measured value. To consider all phase measurements simultaneously in the unwrapping, depending on their respective uncertainty, it is necessary to search for the phase that caused the individual measurements with maximum probability. Since the phase has a periodic structure, the corresponding probability density must be modeled using circular statistics.

The *von Mises* distribution is mathematically easy to handle, it approximates the true distribution of the phase noise quite well, and a maximum-likelihood estimation can be performed in a numerically stable way, cf. Sec. 4.4.3. Therefore, it will be used as the basis for modeling the phase measurement in the following. Modeling using the other densities would work analogously.

The density function of the true phase $\varphi \in [0, 2\pi)$ as a function of the measurement is therefore given by

$$p(\varphi|\varphi_i, \kappa_i) = \frac{e^{\kappa_i \cos(\varphi - \varphi_i)}}{2\pi I_0(\kappa_i)}. \quad (4.38)$$

Here, the measured phase is represented by φ_i and $\kappa_i = 1/\sigma_{\varphi_i}^2$ models the knowledge about the uncertainty of the phase measurement and thus describes the concentration of the distribution. Depending on the frequency of the pattern sequence, the distribution function of the encoded coordinate x can now be derived. With $\varphi_i(x) = 2\pi f_i x$ and with

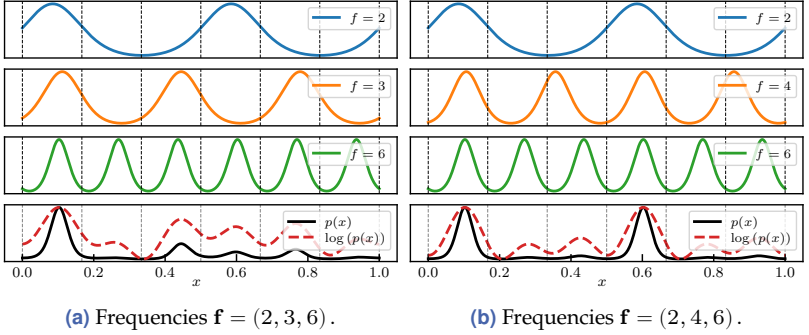


Figure 4.3 Rows 1-3: Multi-modal *von Mises* distributions for different frequencies. Row 4: Compound probability density function and corresponding log-density. (a) The density has a unique maximum, since $\gcd(\mathbf{f}) = 1$. (b) The solution of the maximum-likelihood estimation has a two-fold ambiguity, since $\gcd(\mathbf{f}) = 2 > 1$.

the known frequency f_i , the multi-modal *von Mises* distribution on the periodic interval $x \in [0, 1)$ is obtained:

$$p(x|\varphi_i, \kappa_i, f_i) = \frac{e^{\kappa_i \cos(2\pi f_i x - \varphi_i)}}{I_0(\kappa_i)}. \quad (4.39)$$

Due to the multi-modal character of the distribution, the ambiguity of the phase measurement becomes illustratively visible in the density functions, see figure 4.3.

Since the acquisition of the sinusoidal pattern sequence using phase-shift coding is performed independently for each image and identical acquisition conditions are assumed, each image has in principle the same standard deviation σ_I of the image noise. Therefore, the strength of the phase noise σ_φ remains the same in each measurement. Nevertheless, the variable substitution $\varphi_i(x) = 2\pi f_i x$ reduces the width of the distribution locally by $1/f_i$. This leads to a reduction of the uncertainty, which in turn is bought by an f_i -fold ambiguity.

While the image noise generally remains the same for all images, the estimated phase uncertainty can vary significantly for different situations. For example, if impulse noise appears in images, it is detected by the phase-shift coding as a reduction in the modulation B , which leads to an increase in the estimated uncertainty σ_{φ_i} for the respective pixels. On

the other hand, if the sinusoidal pattern is blurred due to the imaging system, the local contrast of the pattern sequence decreases. Again, the modulation B is affected and the uncertainty increases for the whole phase measurement. Of course, this is strongly influenced by the used pattern frequencies. The uncertainty estimate thus contains knowledge about the system and can therefore be integrated efficiently into the probabilistic modeling of the phase estimate.

Depending on the chosen frequency of the sinusoidal pattern sequence, each phase measurement φ_i corresponds to an individual probability distribution $p(x|\varphi_i, \kappa_i, f_i)$. Since each phase measurement φ_i is measured independently and all have the same underlying coordinate, the compound density of x for given frequencies $\mathbf{f} = (f_1, \dots, f_N)$, phase measurements $\boldsymbol{\varphi} = (\varphi_1, \dots, \varphi_N)$, and estimated concentration parameters $\boldsymbol{\kappa} = (\kappa_1, \dots, \kappa_N)$ can be directly expressed:

$$p(x|\boldsymbol{\varphi}, \boldsymbol{\kappa}, \mathbf{f}) = \prod_i p(x|\varphi_i, \kappa_i, f_i) = \frac{e^{\sum_i \kappa_i \cos(2\pi f_i x - \varphi_i)}}{\prod_i I_0(\kappa_i)}. \quad (4.40)$$

4.4.3 Maximum-Likelihood Phase Unwrapping

Having described the probability density function of the multi-frequency phase-shift coding, this can now be used to find the most likely coordinate that caused the phase measurements. The optimal coordinate and thus the simultaneous unwrapping of all phase measurements can be found with a maximum-likelihood estimator. As a result, maximizing the density function yields the sought coordinate

$$\begin{aligned} \hat{x}_{\text{ML}} &= \arg \max_x p(x|\boldsymbol{\varphi}, \boldsymbol{\kappa}, \mathbf{f}) \\ &= \arg \max_x \log(p(x|\boldsymbol{\varphi}, \boldsymbol{\kappa}, \mathbf{f})) \\ &= \arg \max_x \sum_i \kappa_i \cos(2\pi f_i x - \varphi_i) - \log I_0(\kappa_i) \\ &= \arg \max_x \sum_i \kappa_i \cos(2\pi f_i x - \varphi_i). \end{aligned} \quad (4.41)$$

The logarithm of the Bessel function can be ignored due to its independence of x , and the monotonicity of the logarithm helps to simplify the equations and removes the potentially numerically more unstable exponential function.

4.4.3.1 Uniqueness

To be able to identify a unique maximum, constraints must be applied to the selected frequencies. With other unwrapping methods from the literature, uniqueness can be achieved if the frequencies are relatively prime [149, 255]. However, while all the frequencies need to be pairwise co-prime integers with $\gcd(f_i, f_j) = 1, \forall i \neq j$ for classical number-theoretical approaches, the presented approach has a less restrictive condition. Here, uniqueness is obtained with $\gcd(\mathbf{f}) = \gcd(f_1, f_2, \dots) \leq 1$, where the frequencies do not necessarily need to be integer-valued. For frequencies $f_i \in \mathbb{Q}$, the extension of the gcd to rational numbers can be used to check uniqueness. For frequencies that are irrational numbers, the maximum of (4.41) is theoretically always unique if $\exists f_i \neq f_j$ with $i \neq j$. Though, in this case, when the frequencies are poorly chosen, the unwrapping might be more susceptible to noise. Figures 4.3(a) and 4.3(b) demonstrate the uniqueness constraint illustratively. In figure 4.3(a) the frequencies are set to $\mathbf{f} = (2, 3, 6)$, thus $\gcd(\mathbf{f}) = 1$. Even though, with $\gcd(2, 6) = 2$ and $\gcd(3, 6) = 3$, the frequencies are not pairwise co-prime, a unique maximum of the compound probability density can still be found. In figure 4.3(b) the frequencies are set to $\mathbf{f} = (2, 4, 6)$, thus $\gcd(\mathbf{f}) = 2$. Here the maximum has a two-fold ambiguity. The compound density is only unique in the range $x \in [0, 0.5)$ and repeats itself in $x \in [0.5, 1)$. Thus, in this case, the phase cannot be recovered uniquely.

4.4.3.2 Finding the Maximum

Although (4.41) seems simple, no analytical solution can be given for the global maximum because of the many local extrema. Therefore, the problem must be solved numerically. However, no global optimizer (e.g., simulated annealing, differential evolution) can be used because it could get stuck in a local maximum. To ensure that the maximum of the probability density is found every time and to avoid unwrapping errors, the optimization problem is solved on subintervals. To define the subintervals, (4.41) must be interpreted as a signal $g(x) = \sum_i \kappa_i \cos(2\pi f_i x - \varphi_i)$. Since it is a summation of sinusoidal signals, the maximum frequency of the signal $g(x)$ is equivalent to the maximum used frequency $f_{\max} = \max(f_i)$ in the phase-shift coding. From sampling theory, it is known that a discrete

signal can be reconstructed from its sampling points only if the signal does not change significantly between said points [31]. Consequently, the sampling frequency must be respected. Given the maximum frequency f_{\max} and using the sampling theorem, a minimum required number of intervals $I_{\min} = \lceil 2f_{\max} \rceil$ is obtained in which the global maximum must uniquely lie as a single extremum. A simple 1D line search procedure (see [140]) is now used to find the local maximum in each of those subintervals. A comparison of the local maxima of the intervals finally yields the global maximum.

From a purely practical point of view, it would be sufficient to reduce the interval number to $I_{\min} = \lceil f_{\max} \rceil$, since only the local maxima are required and not the minima. Empirical investigations showed, however, that in rare cases nearly saddle point-like shapes appear in the signal. In these cases, two local maxima can lie very close to each other, and thus, with the reduced number of intervals, only one can be identified as a local maximum in the optimization. Nonetheless, the global maximum could always be found unambiguously in billions of simulations, since the signal changes very strongly in the vicinity of the global maximum, and thus, only a single solution exists in the interval under investigation.

As a remark, it remains to say that the presented maximum-likelihood optimization can in principle also be carried out with the other distributions from Sec. 4.4.1. Though, since the log-likelihood function of the corresponding densities cannot be represented as a simple sum of cosine functions, the spectrum of these log-likelihood functions also has components at higher frequencies. Nevertheless, empirical investigations showed that higher frequencies are attenuated so strongly that the sampling theorem is almost fulfilled and hence a maximum could still be found every time. However, this could only be observed, when all $I_{\min} = \lceil 2f_{\max} \rceil$ subintervals were searched. Thus, the other density functions need twice the computation time as compared to the *von Mises* distribution.

In summary, with the presented method, the wrapping of all phases is compensated simultaneously and all measurements are fused to an optimal solution so that finally the most likely value of the coordinate x can be found.

4.4.4 Spatio-Temporal Phase Unwrapping

Temporal phase unwrapping has the great advantage that each pixel can be individually unwrapped and an absolute phase is obtained. This is especially useful when only a little information about the surface to be examined is known and when 2D unwrapping methods would lead to erroneous results. In many tasks of optical metrology, where structured illumination is used, continuous surfaces are often examined. For example, deflectometry often works with lacquered body parts from the automotive industry, with lenses, or parabolic mirrors, which can be described for the most part as continuous surfaces with only a few regions deviating from this continuity due to sharp edges. Also in time-of-flight imaging and many areas of profilometry, *i.e.*, fringe projection, piecewise continuous objects are often inspected [47, 209]. This piecewise continuity has the consequence that neighboring camera pixels will observe similar phase values on the surface. It is therefore reasonable to use this additional information to help with the phase unwrapping to suppress phase errors.

The assumption of local continuity should be integrated into the probabilistic framework from the previous section. This allows performing not only an unwrapping in the temporal dimension but a 3D phase unwrapping while implicitly smoothing the probability density functions over the spatial dimensions. To do this, the probability density of each camera pixel is modeled as a superposition of the probability densities of the local neighborhood. The probability density for each individual pixel \mathbf{u} was already derived in the previous section and can be considered as a conditional density

$$p(x(\mathbf{u})|\boldsymbol{\varphi}(\mathbf{u}), \boldsymbol{\kappa}(\mathbf{u}), \mathbf{f}) = \prod_i p(x(\mathbf{u})|\varphi_i(\mathbf{u}), \kappa_i(\mathbf{u}), f_i). \quad (4.42)$$

If neighboring pixels can no longer be considered independently of each other, then the probability density results in a weighted superposition of individual densities for each pixel \mathbf{u}

$$p(x(\mathbf{u})) := \sum_{\hat{\mathbf{u}} \in \mathcal{U}(\mathbf{u})} p(\mathbf{u}|\hat{\mathbf{u}}) p(x(\hat{\mathbf{u}})|\boldsymbol{\varphi}(\hat{\mathbf{u}}), \boldsymbol{\kappa}(\hat{\mathbf{u}}), \mathbf{f}), \quad (4.43)$$

where $\mathcal{U}(\mathbf{u})$ represents a set of relevant neighborhood pixels. Since more distant pixels have less influence and the modeling should be approached

with minimal knowledge about the observed surface, the transition probabilities are modeled using a 2D normal distribution

$$p(\mathbf{u}|\hat{\mathbf{u}}) = \mathcal{N}(\hat{\mathbf{u}}, \sigma_N^2 \mathbf{I}) = \frac{1}{2\pi\sigma_N^2} e^{-\frac{\|\mathbf{u}-\hat{\mathbf{u}}\|^2}{2\sigma_N^2}}. \quad (4.44)$$

The compound density, consisting of a spatial modeling by means of normal distributions and a temporal modeling by means of *von Mises* distributions, finally results in

$$p(x(\mathbf{u})) = \sum_{\hat{\mathbf{u}} \in \mathcal{U}(\mathbf{u})} \frac{\exp\left(-\frac{\|\mathbf{u}-\hat{\mathbf{u}}\|^2}{2\sigma_N^2}\right) \exp\left(\sum_i \kappa_i(\hat{\mathbf{u}}) \cos(2\pi f_i x - \varphi_i(\hat{\mathbf{u}}))\right)}{2\pi\sigma_N^2 \prod_i I_0(\kappa_i(\hat{\mathbf{u}}))}. \quad (4.45)$$

Although this probability density appears more complicated than the equation (4.40) from the previous section, it can be maximized using the same methods for finding the optimal solution of the coordinate: $\hat{x}_{\text{ML}}(\mathbf{u}) = \arg \max p(x(\mathbf{u}))$.

However, it must be considered that this approach only leads to meaningful results if the local continuity assumption is not violated. To ensure that the given model is only applied in continuous areas, discontinuities have to be detected.

4.4.4.1 Detection of Discontinuities

Depending on the application, discontinuities in a surface can lead to discontinuities in the phase map. In the case of profilometry, a step in the surface results in a step in the phase map, whereas a step in the surface gradient does not necessarily destroy the continuity of the phase map. However, in the case of a deflectometric measurement of specular surfaces, even a step in the surface gradient may result in a step in the phase map. Consequently, this means that it is not the intention to detect edges on the surface but discontinuities in the unwrapped phase.

For edge detection, a simple detector operating directly on the wrapped phase estimates is suitable for this purpose. Nonetheless, since the 2π -discontinuities contained in the phase maps do not represent a property of the surface, they must not be falsely detected, cf. figure 4.10. Thus, a 2π -invariant detector is needed. Typically, gradient-based operators

are utilized to detect edges in images. For this, the Laplace operator $\Delta\varphi(\mathbf{u}) = \partial_{s^2}\varphi(\mathbf{u})^2 + \partial_{t^2}\varphi(\mathbf{u})^2$ is often used. However, for a 2π -phase jump in the wrapped phase, the operator will yield a multiple of 2π even when the correctly unwrapped phase would have only a small continuous change. To have this property ignored, a 2π -invariant Laplace operator is defined

$$\Delta_{2\pi}\varphi(\mathbf{u}) := \Delta\varphi(\mathbf{u}) \bmod 2\pi, \quad (4.46)$$

which is only sensitive to phase discontinuities in the unwrapped phase caused by the surface, whereas discontinuities that are caused by the ambiguity of a wrapped phase are ignored. To reduce the effect of noise in edge detection, a Laplacian of Gaussian may be used. Equation (4.46) can take values within the periodic interval $[0, 2\pi)$. However, since the strength of an edge is defined as the distance to 0, it is necessary to calculate the circular distance for an appropriate edge quality measure. Hence, for every phase measurement $\varphi_i(\mathbf{u})$, an energy measure

$$E_i(\mathbf{u}) = d^\circ(0, \Delta_{2\pi}\varphi_i(\mathbf{u})) = \pi - |\pi - |\Delta_{2\pi}\varphi_i(\mathbf{u})|| \quad (4.47)$$

is calculated in which the maximum possible circular distance is equal to π , which would correspond to a strong edge feature. Further, an appropriate averaging over all phase maps improves the edge estimate

$$E(\mathbf{u}) = \frac{\sum_i \sigma_{\varphi_i}^{-2}(\mathbf{u}) E_i(\mathbf{u})}{\sum_i \sigma_{\varphi_i}^{-2}(\mathbf{u})}, \quad (4.48)$$

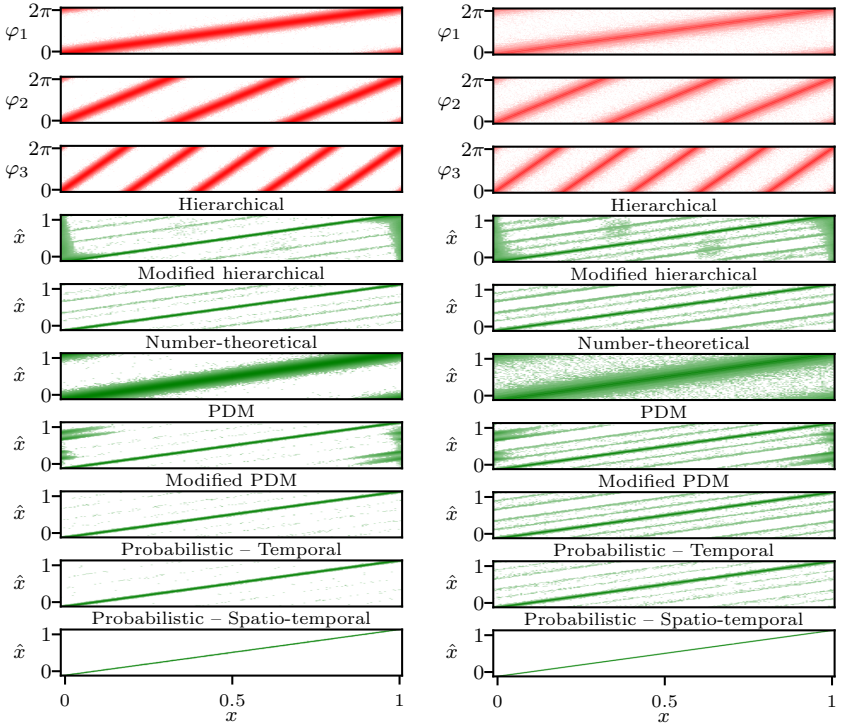
where the uncertainty of the phase estimate can be taken into account. Hence, the application of the modified Laplacian operator ultimately provides an energy measure for an edge, which is insensitive towards 2π -discontinuities. And finally, subsequent thresholding on this energy measure results in a feature map containing edge areas and non-edge areas, see figure 4.10. In places where an edge has been detected, the temporal modeling according to Sec. 4.4.2 must be used, whereas everywhere else the modeling according to Sec. 4.4.4 may be used to improve the phase unwrapping by utilizing the spatial neighborhood information.

4.5 Evaluation

In this section, the presented methods are evaluated, analyzed, and compared with the state of the art. Sinusoidal pattern sequences with different frequencies are simulated and the respective phase is estimated using phase-shift coding, where the number of steps is chosen to be $M = 8$. The following unwrapping methods are examined: The hierarchical method of Huntley and Saldner [88], the proposed modified hierarchical method, the heterodyne method of Lai *et al.* [107], the number-theoretical method of Towers *et al.* [202], the PDM method of Zuo *et al.* [255], the proposed modified PDM method, the proposed probabilistic temporal method, and the proposed probabilistic spatio-temporal method. For the proposed probabilistic methods, the *von Mises* probability density is used, unless specified otherwise. For the spatio-temporal method a spatial neighborhood $\mathcal{U}(\mathbf{u})$ of 3×3 pixels is used. To investigate the robustness of the presented phase unwrapping algorithms, the influence of Gaussian image noise and impulse noise is examined.

4.5.1 Qualitative Comparison

The resolution of the reference pattern generator was set to (2003, 2003). For the first simulation three phase measurements with frequencies $\mathbf{f} \approx (1, 3, 5)$ were generated. Because for the number-theoretical method pairwise co-prime wavelengths must be used, the wavelengths are quantized as $\boldsymbol{\lambda} = (2003, 668, 401)$. This corresponds to the set of frequencies $\mathbf{f} \approx (1, 2.999, 4.995)$. Nevertheless, since no methods are restricted to integer frequencies, this does not result in any major disadvantages. The phase uncertainty was chosen to be $\sigma_\varphi = 0.25 \text{ rad} = 14.3^\circ$. Using (4.10), Gaussian noise with variance $\sigma_1^2 = \sigma_\varphi^2 B^2 M / 2$ was added to the sinusoidal pattern sequence. It is important to note that the noise is not added to the wrapped phase measurements, as it is often done in the literature, but to the camera images I_m , otherwise no realistic statements about phase-shift coding can be made. The heterodyne method calculates a phase difference to obtain a unique reference phase. Since φ_1 is already unique, it does not make sense to evaluate the heterodyne method for this frequency configuration. The coordinate $x \in [0, 1)$ was sampled in 2003 steps and each value was simulated 2003 times. Figure 4.4(a)

(a) Gaussian noise with $\sigma_\varphi = 0.25$ rad.

(b) Impulse noise in 15% of the pixels.

Figure 4.4 Top: Noisy phase measurements φ_i with frequencies $\mathbf{f} \approx (1, 3, 5)$. Bottom: Estimated coordinate \hat{x} for different methods.

shows the phase measurements and the coordinates estimated with the different unwrapping methods. Here, stronger colors represent a higher point density. The upper three plots show the noisy phase measurements φ_i over the true coordinate x . The lower plots show the corresponding estimated coordinates \hat{x} over x .

The hierarchical unwrapping shows a line of correctly unwrapped estimates in the middle section. At the boundaries of the coding interval, large errors appear because the periodicity of the phase is not implicitly modeled for this method. For these reasons, the effective coding interval is often reduced in practical applications. This avoids unwrapping errors,

but the effectively used frequency decreases, which increases the overall phase uncertainty. In addition to the boundary errors, the hierarchical method shows unwrapping errors that are represented by parallel lines to the middle line. In these cases, the phase was incorrectly unwrapped once or even twice. Since the hierarchical method always refers back to the unwrapped previous phase, unwrapping errors propagate from top to bottom and can no longer be compensated once they occurred. Using the modified hierarchical method, boundary errors can be significantly reduced, due to respecting the circularity of the phase by using the circular mean. Similar to the standard hierarchical method, parallel lines of unwrapping errors appear, since the second phase is unwrapped by only using the first measurement. The number-theoretical unwrapping shows no errors at the boundary of the coding interval since the Chinese remainder theorem is based on modulo arithmetic. Also, only a few unwrapping errors occur in the middle of the coding interval. Two lines of faulty estimations appear near the boundary, where noisy estimates $x > 1$ and $x < 0$ are folded back into the used interval $[0, 1)$. Overall, the method is more susceptible to noise, which results in a coordinate estimation with greater uncertainty. The PDM unwrapping is much better. Almost all pixels in the middle of the coding interval are unwrapped correctly. The errors at the boundary are caused by the lack of modeling of the periodicity of the phase. By using the modified PDM method, these wrongly unwrapped pixels can be corrected. The proposed probabilistic temporal method can also compensate for the boundary errors since the periodicity is well described using circular statistics and it performs satisfactorily in other areas as well. Only a few pixels are unwrapped incorrectly. Finally, the spatio-temporal method yields even better results. Here almost all values are unwrapped correctly and the uncertainty of the estimation is the smallest compared to the other methods, as can be seen by the overall thinner line. In other words, it makes a lot of sense to include spatial information.

In a second simulation, the sinusoidal pattern sequence is superimposed with impulse noise, where the probability of an impulse is set to $p_I = 0.15$. An impulse in the image appears either as a black pixel or as a white pixel, *i.e.*, it acts like salt and pepper noise. Again, of course, the noise must be added to the sinusoidal pattern sequence I_m and not to the wrapped phase maps. Although 15% of the pixels show an impulse,

the same number of phase estimates may not necessarily be affected. Figure 4.4(b) shows the phase measurements and the coordinates estimated with the different unwrapping methods. An impulse in the pattern sequence causes the respective phase to be distorted to a greater or lesser extent, depending on how far the impulse is from the correct intensity value.

Again, the hierarchical method shows similar effects as before, which are, however, more prominent here. The modified hierarchical method is slightly better than the standard approach and also the errors at the boundaries are smaller. The number-theoretical method can still provide a good unwrapping performance. Due to the generally higher noise level, the accuracy decreases. For the PDM method, more erroneous estimations occur, comparable to the hierarchical method, whereas the modified PDM method reduces the boundary errors. As opposed to this, the two probabilistic methods can still achieve very good results. This can be explained by considering that a phase measurement with an impulse in the pattern sequence has a smaller modulation B . This also increases the corresponding estimate of the phase uncertainty. This estimate can be used directly in the proposed methods to compensate for poor phase measurements. Thus, better phase measurements have more influence on the optimization. For the spatio-temporal method, this means that a distortion of the phase has an effect only if a large number of the pixels in the respective $3 \times 3 \times 3$ cube is disturbed. Since the probability of this is quite low, the method yields almost no errors.

To evaluate the heterodyne method, phase measurements with frequencies $\mathbf{f} \approx (6, 9, 11)$ are generated. For the same reasons as before, the wavelengths were quantized as $\boldsymbol{\lambda} = (331, 223, 181)$. This corresponds to frequencies $\mathbf{f} \approx (6.051, 8.982, 11.066)$. Image noise is superimposed on the sinusoidal pattern sequence, corresponding to a phase uncertainty of $\sigma_\varphi = 0.15 \text{ rad} = 8.6^\circ$. Since the hierarchical and modified hierarchical method can uniquely unwrap the phases only up to the first period, they are not considered in this comparison. Figure 4.5(a) shows the phase measurements and the coordinates estimated with the different methods.

It can be seen that even with smaller noise than before, the heterodyne method delivers only mediocre results. A large part of the pixels is unwrapped correctly, though many lines of incorrect values appear parallel to the correct line. This is due to the fact that the phase noise

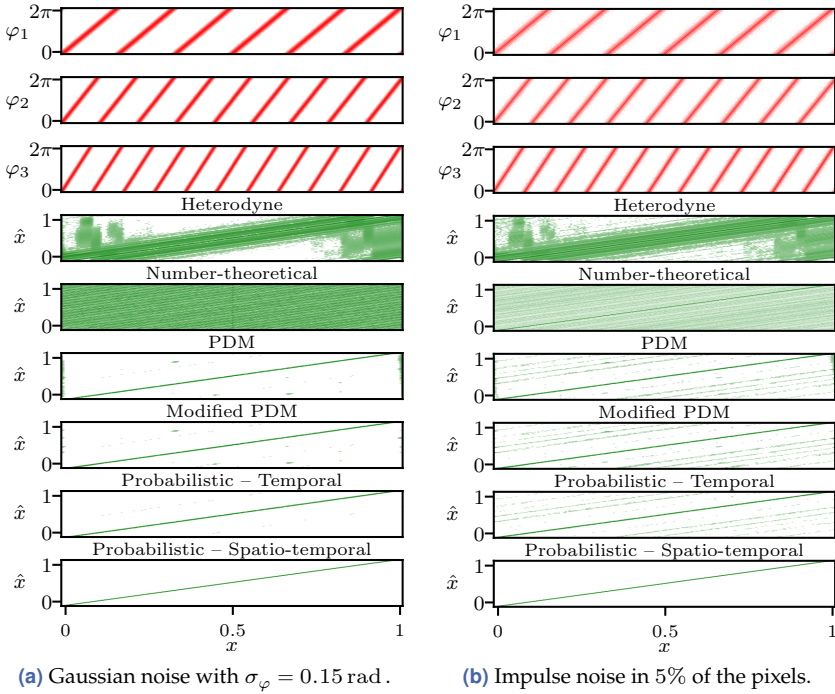


Figure 4.5 Top: Noisy phase measurements φ_i with frequencies $\mathbf{f} \approx (6, 9, 11)$. Bottom: Estimated coordinate \hat{x} for different methods.

is summed up when calculating the phase difference. To get a unique solution, first $\varphi_{12} = \varphi_1 - \varphi_2$ with $f_{12} = f_2 - f_1 \approx 2.93$ and φ_{23} with $f_{23} = 2.08$ are calculated. A unique phase can then be calculated with $\varphi_{123} = \varphi_{23} - \varphi_{12}$ with $f_{123} = f_{12} - f_{23} = 0.85$. This is then used to unwrap the individual phase measurements. However, since the noise is summed up in each step, the reference phase is of poor quality, resulting in a poor overall unwrapping result. Surprisingly, the number-theoretical method fails completely. The integer arithmetic of the method cannot work even at a very small noise level. The PDM method and the modified PDM method show almost the same very good result, with only a few boundary errors and two small clusters of erroneous estimates. As before,

the presented probabilistic methods provide very good results, whereas the spatio-temporal approach yields almost only correct estimates.

For a final evaluation, the sinusoidal pattern sequence was now again overlaid with impulse noise, with the probability of an impulse set to $p_1 = 0.05$. Figure 4.5(b) shows the phase measurements and the coordinates estimated with the different methods. Although the noise is very small, the heterodyne method again shows many unwrapping errors. The number-theoretical method delivers bad values too. Here we see that almost only pixels without distortion are unwrapped correctly, visible by the somewhat stronger line in the center. The PDM method can unwrap the phase very well, as before. Errors are still found at the boundary and in parts in the center, whereas the modified version better compensates for the boundary errors. The presented probabilistic methods show almost perfect results, which is explainable by the incorporation of the estimated phase uncertainty in the unwrapping process.

4.5.2 Robustness Against Noise

The methods presented are now being evaluated quantitatively. For this purpose, the robustness of the methods against Gaussian noise and impulse noise will be investigated. In order to compare all methods, sinusoidal pattern sequences with $M = 8$ phase shifts were simulated. Subsequently, various noise factors were superimposed on the images, the phase was estimated using phase-shift coding, and finally, the phases were unwrapped using the presented methods.

4.5.2.1 Error Metrics

In order to make quantitative statements about the methods, suitable error metrics have to be defined beforehand. As a first error measure, the estimation error

$$\epsilon_x = |x - x_{\text{true}}| \quad (4.49)$$

defines the absolute distance of the estimated coordinate x to the true coordinate x_{true} . The second error metric evaluates the quality of phase

unwrapping and describes the success rate, representing whether a pixel was correctly unwrapped:

$$s_x = \frac{1}{N} \sum_{i=0}^{N-1} C_i, \quad (4.50)$$

where C_i indicates whether the phase measurement associated with the frequency f_i has been correctly unwrapped

$$C_i = \begin{cases} 1, & |k_{i,\text{true}} - k_i| = 0 \\ 0, & \text{otherwise} \end{cases}, \quad (4.51)$$

$$= \begin{cases} 1, & \frac{1}{2f_i} > |x_{\text{true}} - \frac{1}{2\pi}\Phi_i| \\ 0, & \text{otherwise} \end{cases}. \quad (4.52)$$

Because the proposed methods do not directly unwrap the individual phase measurements but return a global solution, (4.52) is used with $C_i = C_{i_{\max}}$ and $i_{\max} = \arg_i \max f_i$. Hence, any phase value that is farther away from the true solution than $1/(2f_{\max})$ is therefore classified as an unwrapping error.

4.5.2.2 Error Evaluation

For a first analysis, the frequencies of the sinusoidal pattern sequence were again chosen to be $\mathbf{f} \approx (1, 2.999, 4.995)$ to create integer wavelengths $\lambda = (2003, 668, 401)$ to ensure that the number-theoretical method can be used. The robustness towards Gaussian image noise was analyzed by increasing the phase uncertainty incrementally from $\sigma_\varphi = 0$ to $\sigma_\varphi = 0.5 \text{ rad} \approx 28.6^\circ$ in 100 steps. For the analysis of robustness to impulse noise, the probability of an impulse was increased stepwise from $p_I = 0$ to $p_I = 20\%$ in 100 steps. Figure 4.6 shows the results of the analysis as a plot of the mean estimation error ϵ_x and mean success rate s_x .

The evaluation of the phase error metrics yields similar results as the evaluation of the qualitative results from the previous section, for both Gaussian noise and impulse noise. When analyzing the influence of Gaussian noise, large differences between the methods can be observed. The number-theoretical method consistently yields the worst results with

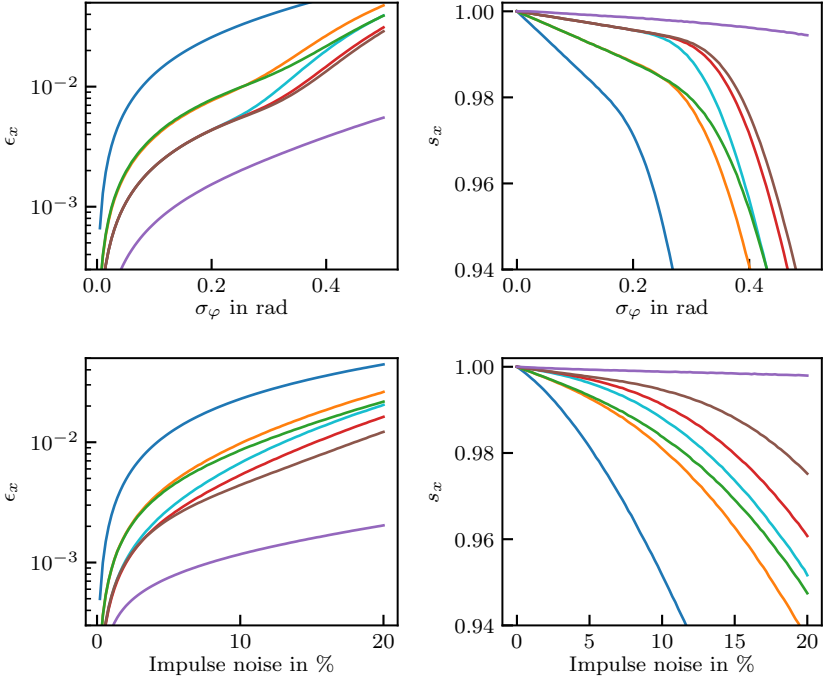


Figure 4.6 Evaluation of phase error ϵ_x and success rate s_x with $\mathbf{f} \approx (1, 3, 5)$ for different phase unwrapping methods: — Number-theoretical, — hierarchical, — modified hierarchical, — PDM, — modified PDM, — probabilistic (temporal), — probabilistic (spatio-temporal).

the largest estimation error. The success rate is also consistently the lowest, mainly caused by the erroneous unwrapping at the boundaries of the coding interval. Interestingly, the hierarchical method and the PDM method show almost identical behavior up to about $\sigma_\varphi \approx 0.2$. Only for higher noise levels, the advantage of the PDM method becomes apparent, resulting in a lower estimation error and a higher success rate. The modified hierarchical method and the modified PDM method show the same behavior as the probabilistic temporal method for lower noise levels. For high noise levels, the modified hierarchical method becomes comparable to the standard PDM method. The proposed probabilistic methods provide the best results with the smallest estimation error and highest

success rate, where even for very large noise levels the spatio-temporal method can still correctly unwrap more than 99.9 % of the pixels. The same conclusions can be made for the analysis of impulse noise. Here it becomes even possible to order the methods from worst to best directly by looking at the plots: number-theoretical, hierarchical, PDM, modified hierarchical, modified PDM, probabilistic temporal, probabilistic spatio-temporal.

For the second analysis, the frequencies of the sinusoidal pattern sequence were again chosen to be $\mathbf{f} \approx (6.051, 8.982, 11.066)$ to create integer wavelengths $\boldsymbol{\lambda} = (331, 223, 181)$ suitable for the number-theoretical method. The noise was parameterized in the same way as before. Figure 4.7 shows the results of the analysis as a plot of the mean phase error ϵ_x and mean success rate s_x . While analyzing the influence of Gaussian noise, it can be seen that the number-theoretical method is extremely susceptible to noise. It can only deliver correct values for very small noise values. Starting from a noise of $\sigma_\varphi \approx 0.02$ it has already reached the maximum possible mean error. For small noise levels, the heterodyne method still shows very good results and can keep up with the other methods. Only for larger noise, significant deficiencies become apparent. For the investigated frequency configuration, the standard and the modified PDM method have an almost identical success rate, which is only slightly worse as compared to the probabilistic temporal method. Also, the probabilistic method is slightly better for low noise levels resulting in a smaller estimation error. For large noise levels, all yield almost the same result. The spatio-temporal method, on the other hand, still yields very good results for high noise levels even when a phase-shift configuration is used consisting of high frequencies, where in general the success rate is more susceptible to noise.

The analysis of the impulse noise emphasizes again the advantages of the proposed methods. The number-theoretical method and the heterodyne method are very susceptible to impulse noise. Even small amounts of noise cause the success rate to drop steeply and the estimation error to rise significantly. The PDM method and the modified PDM method show similar behavior, with the modified method being slightly better. Again, the probabilistic temporal method gives better results than the classical approaches for all noise levels. Interestingly, the spatio-temporal method

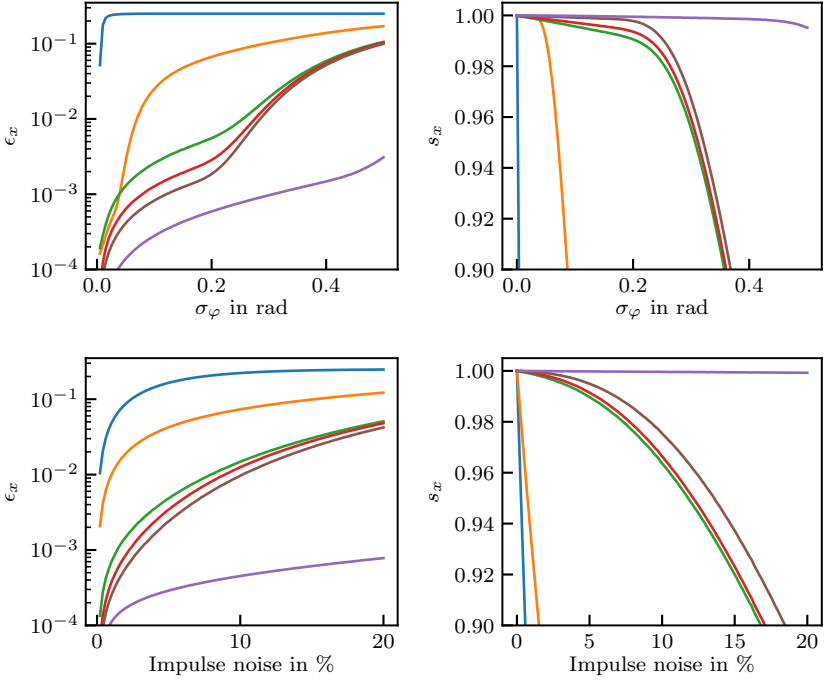


Figure 4.7 Evaluation of phase error ϵ_x and success rate s_x with $\mathbf{f} \approx (6, 9, 11)$ for different phase unwrapping methods: — Number-theoretical, — heterodyne, — PDM, — modified PDM, — probabilistic (temporal), — probabilistic (spatio-temporal).

shows exceptionally good results here. Even with $p_I = 20\%$ impulse noise, the success rate is still greater than 99.99%. This can be explained by the fact that a coordinate estimation is only disturbed if a certain number of phase measurements are influenced by an impulse. Since the spatio-temporal method combines 27 probability densities for each coordinate estimate, the probability that a large part of these densities is disturbed is very small. To obtain a correct estimate, at least one pixel of the 3×3 spatial neighborhood must be correct for only two of the three phase measurements, since the corresponding frequencies are pairwise co-prime and effectively two phase measurements are sufficient to get a unique result. The probability of an unwrapping error at $p_I = 20\%$

impulse noise with a spatial neighborhood of $S = 9$ pixels, $N = 3$ frequencies, and pairwise co-prime frequencies is therefore approximately

$$1 - \sum_{n=2}^N \binom{N}{n} (1 - p_I^S)^n (p_I^S)^{N-n} = N(1 - p_I^S) (p_I^S)^{N-1} + (p_I^S)^N \quad (4.53)$$

$$= 0.2^{27} + 3 \cdot 0.2^{18} (1 - 0.2^9) \approx 6.3 \cdot 10^{-13}.$$

The probability may be even lower since not every impulse necessarily causes an erroneous measurement.

4.5.3 Comparison of Different Phase Noise Models

To confirm the choice of the *von Mises* distribution as a representative for the probability density of the phase, this section compares the different densities.

4.5.3.1 Robustness against Model Errors

The phase uncertainty is in principle not known but has to be estimated by using the standard deviation σ_I of the underlying image noise. However, since this is either set arbitrarily or has to be estimated from the camera parameters, model errors may be introduced. To investigate the robustness against these model errors, a phase measurement with image noise $\sigma_I = 0.3$ is simulated. The different probability densities are parameterized with the incorrect $\tilde{\sigma}_I = \delta_{\sigma_I} \sigma_I$ where the relative deviation δ_{σ_I} describes the model error. Figure 4.8 shows the influence of the model error on the temporal and spatio-temporal phase unwrapping. For the temporal approach, the log-likelihood of the *von Mises* function is used. Here, the phase unwrapping is completely independent of the model error. Because the image noise has only a multiplicative influence on the estimated phase uncertainty, this factor can be extracted from the objective function (4.41) and has no significant influence on the maximization. However, when the other distributions or the spatio-temporal approach is used, the situation is different. Here, the influence of model errors as well as numerical instabilities become apparent. For $\delta_{\sigma_I} < 0.5$ the error for the *von Mises* density and the phase-shift model becomes

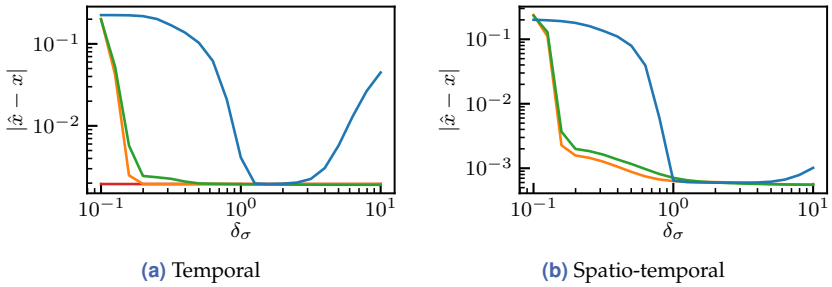


Figure 4.8 Evaluation of model deviations for different probability density functions: — Wrapped Gaussian, — *von Mises*, — *von Mises* (log-likelihood), — Phase-shift model.

larger and sometimes the optimization of the phase-shift model fails so that no result can be obtained. For too small uncertainties, the terms in the exponential functions of the density (4.37) become too large, and thus numerically bad or even invalid values can occur. The effects are even stronger for the spatio-temporal approach. The wrapped Gaussian density, on the other hand, shows good results only starting from $\delta\sigma_1 = 1$, whereas the results deteriorate again starting from $\delta\sigma_1 \approx 2$. In summary, the log-likelihood approach is completely robust to model errors, whereas the *von Mises* density and the phase-shift model show poor results only for very small values. To avoid numerical instabilities, the assumed image noise σ_1 should therefore have a lower bound, since it does not have a significant influence on the result. Nevertheless, the relative difference of the phase uncertainty σ_φ and the influence of the frequencies are of course still important.

4.5.3.2 Robustness against Noise

With the optimal σ_1 selected, the probabilistic temporal phase unwrapping is now analyzed in more detail. Table 4.1 shows the estimation error and the success rate for Gaussian image noise with $\sigma_\varphi = 0.3$ and impulse noise with $p_I = 0.03$ for different probability densities. For reference, the PDM method is shown too. As expected, the phase-shift model according to Rathjen [167] gives the best results for the Gaussian noise and the *von Mises* distribution the second best. Compared to the PDM method, the probabilistic methods differ only minimally. Interestingly, for impulse

Table 4.1 Comparison of different phase noise models.

	Method	$\mathbf{f} = (1, 3, 5)$		$\mathbf{f} = (6, 9, 11)$	
		$10 \cdot \epsilon_x$	s_x in %	$10 \cdot \epsilon_x$	s_x in %
Gaussian	PDM	0.772	99.050	1.199	95.457
	Wrapped normal	0.453	99.490	1.032	95.658
	<i>von Mises</i>	0.438	99.526	0.912	96.275
	Phase-shift-model	0.425	99.549	0.862	96.505
Impulse	PDM	0.157	99.839	0.148	99.514
	Wrapped normal	0.086	99.928	0.058	99.812
	<i>von Mises</i>	0.086	99.928	0.059	99.811
	Phase-shift-model	0.091	99.925	0.068	99.781

noise, the wrapped normal distribution performs better than the model of Rathjen. Again, the *von Mises* distribution provides the second-best results, which is only insignificantly worse than the wrapped normal distribution. So, apart from the other advantages of the *von Mises* distribution, it therefore turns out to also be a good compromise to be robust against Gaussian and impulse noise.

4.5.4 Phase Map Reconstruction

This section shows how phase maps are reconstructed using the presented unwrapping methods. For this purpose, two phase maps (512×512 pixels) are generated, see figure 4.9. Phase map 1 shows a continuous surface with hills and valleys, whereas phase map 2 represents a discontinuous surface that has sharp edges. The corresponding sinusoidal pattern sequences are generated with wavelengths $\lambda = (331, 223, 181)$, corresponding to frequencies $\mathbf{f} \approx (6.051, 8.982, 11.066)$. The pattern images are superimposed with Gaussian noise corresponding to $\sigma_\varphi = 0.15$. Figure 4.9 shows the generated phase-shift image data and the wrapped phase maps that are calculated using phase-shift coding.

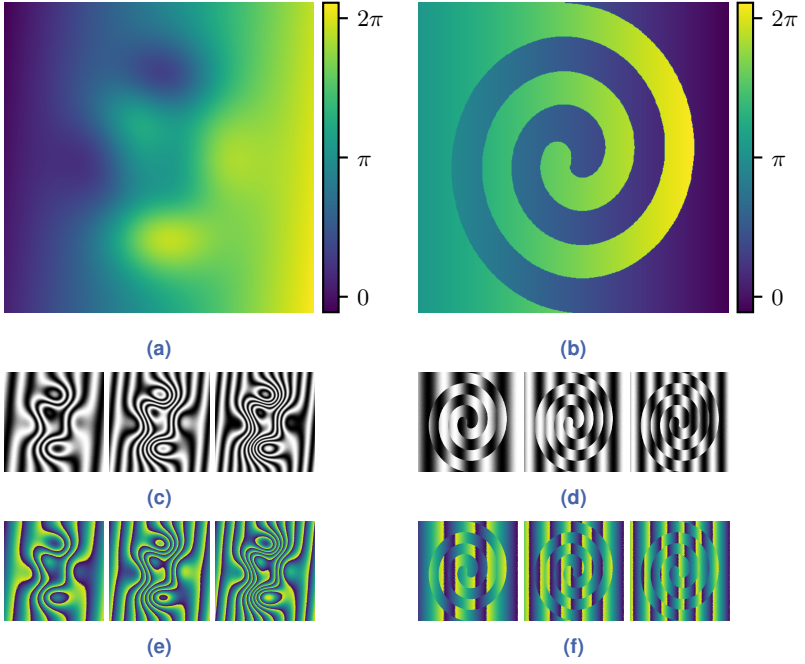


Figure 4.9 (a) & (b) show the true phase maps. (c) & (d) show the noisy sinusoidal patterns for phase offset $\Psi = 0$ for the frequencies $\mathbf{f} \approx (6, 9, 11)^T$, from left to right respectively. (e) & (f) show the corresponding noisy phase maps with phase noise $\sigma_\varphi = 0.15$.

4.5.4.1 Edge Detection Example

Because the presented spatio-temporal method may not be used across discontinuities, edges in the phase map must be detected first. The application of edge-detection to the phase maps is shown in figure 4.10. Figure 4.10(a) and figure 4.10(c) each show the application of the presented 2π -invariant edge detector to the wrapped phase maps. Figure 4.10(b) and figure 4.10(d) show the output of an edge detector that uses a standard Laplacian and a standard absolute distance to obtain the edge instead of the proposed 2π -invariant operations. It can be seen that the standard detector not only detects the edges in the phase map but also the phase jumps caused by the wrapping of the phase values. The presented detec-

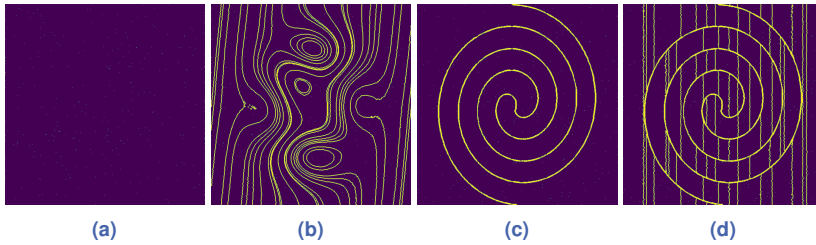


Figure 4.10 Edge detection in phase maps: (a) & (c) show the proposed edge detection for phase maps 1 and 2, respectively. (b) & (d) show the result of a standard edge detection.

tor, on the other hand, detects only the real edges in the phase map. Since phase map 1 is continuous, no edge is detected. Only a few individual pixels are detected as edges since the edge detector is of course also influenced by the noise. The spiral shape of phase map 2 can be detected very well and in addition, only a few individual pixels are incorrectly detected as an edge. An optimization of the thresholding parameter in the edge detection could resolve those wrongly detected pixels. However, even a wrongly detected edge may not cause a faulty phase unwrapping, since edge pixels are then “just” unwrapped using the probabilistic temporal method that instead of the spatio-temporal method still performs very well.

4.5.4.2 Phase Reconstruction

The results of the unwrapping of phase map 1 are shown in figure 4.11 for the heterodyne unwrapping, the PDM unwrapping, the proposed temporal unwrapping, and the proposed spatio-temporal unwrapping, respectively. The top row shows the reconstruction as a 3D plot, with the linearly increasing phase ramp subtracted for better visibility. The middle row shows the reconstructed phase and the bottom row shows the respective error.

It can be seen that the heterodyne method works only suboptimally. The total error is quite high and only 78.19 % of the pixels are correctly unwrapped. The reconstructed phase map looks very noisy. The PDM method, on the other hand, yields 99.89 % correct pixels and thus provides a far smoother phase reconstruction. Single unwrapping errors oc-

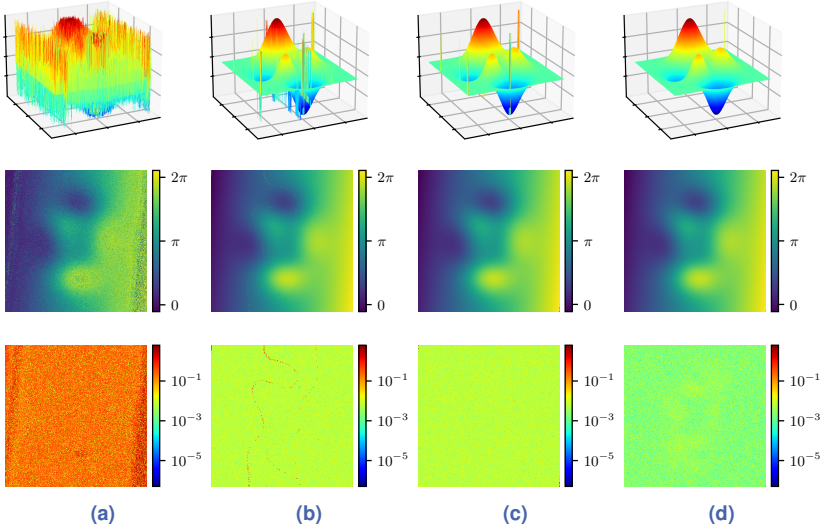


Figure 4.11 Reconstruction of phase map 1 influenced by Gaussian noise with $\sigma_\varphi = 0.15$ rad. The top row shows the phase reconstruction as 3D plot, where the linear phase ramp is removed. The middle row shows the reconstructed phase. The bottom row shows the phase error. (a) Heterodyne: $\epsilon_x = 0.399$, $s_x = 78.19\%$. (b) PDM: $\epsilon_x = 0.011$, $s_x = 99.89\%$. (c) Probabilistic temporal: $\epsilon_x = 0.008$, $s_x = 99.99\%$. (d) Probabilistic spatio-temporal: $\epsilon_x = 0.003$, $s_x = 100\%$.

cur for phase values close to 0 and 2π , e.g., near the large hill and the deep valley. In addition, some unwrapping errors occur in the center of the phase map near lines where the wrapped phases show 2π -discontinuities. Initially, these errors cannot be explained directly. However, as indicated by Petković *et al.* [149], their PDM method performs worse for non-integer frequencies, which therefore could be the cause. The proposed probabilistic temporal method can correctly unwrap 99.99 % of the pixels. Similar to the PDM method, isolated errors occur for values near the boundaries of the coding interval. The errors along the 2π -discontinuities of the wrapped phases do not occur here and show that the proposed method also works properly for rational frequencies. The proposed probabilistic spatio-temporal method can correctly unwrap all pixels. At the same time, the general accuracy is higher, as can be seen in the error map by the overall darker green color. Thus, the local information used in the

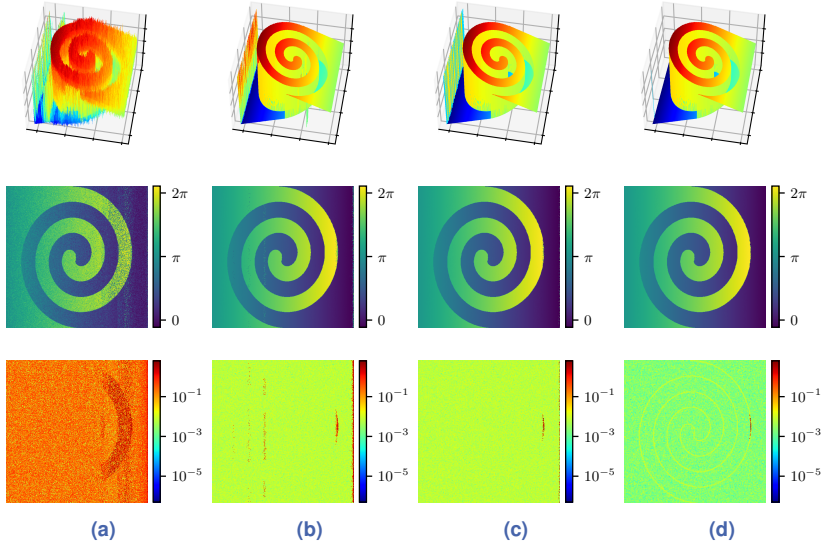


Figure 4.12 Reconstruction of phase map 2 influenced by Gaussian noise with $\sigma_\varphi = 0.15$ rad. The top row shows the phase reconstruction as 3D plot. The middle row shows the reconstructed phase. The bottom row shows the phase error. (a) Heterodyne: $\epsilon_x = 0.418$, $s_x = 78.35\%$. (b) PDM: $\epsilon_x = 0.031$, $s_x = 99.24\%$. (c) Probabilistic temporal: $\epsilon_x = 0.014$, $s_x = 99.90\%$. (d) Probabilistic spatio-temporal: $\epsilon_x = 0.005$, $s_x = 99.97\%$.

maximum-likelihood estimation not only improves the success rate of the unwrapping but also acts as a denoising filter and therefore leads to lower uncertainty in the estimated coordinate.

Figure 4.12 shows the results of the unwrapping of phase map 2, again, for the heterodyne unwrapping, the PDM unwrapping, the presented temporal and spatio-temporal unwrapping, respectively. Here again, the heterodyne method performs significantly worse than the other methods. Only 78.35 % of the pixels can be unwrapped correctly and the estimation error is very high. As before, the PDM method shows errors at the boundaries of the coding interval, which appear at the right edge of the spiral and the right side of the phase map. In addition, unwrapping errors occur near the 2π -discontinuities of the wrapped phases, which could be caused by the non-integer frequencies. The probabilistic temporal method again shows smaller errors at the boundaries of the

coding interval and faulty lines as in the PDM method do not appear. The spatio-temporal method again shows an overall smaller error and can correctly unwrap almost all pixels. The error map shows that the pixels along the edge of the spiral have a larger error. This can be explained by the fact that for these pixels the continuity assumption of the surface is violated and these pixels were detected as an edge, see figure 4.10(c). Wherever an edge is detected, the temporal method is used, everywhere else the spatio-temporal method helps to improve the estimation. Further, the spatio-temporal method is more robust against unwrapping errors, which can also be seen in the error map at the right edge of the spiral. Here, only unwrapping errors occur exactly on the edge. The pixels away from the edge can be correctly unwrapped.

4.6 Summary

This chapter aimed to find a way to measure the deflectometric imaging function, which is needed for deflectometry as well as for the camera calibration presented in this thesis. An optical encoding utilizing phase-shift coding was discussed, which allows finding a direct mapping of camera pixels to points in the plane of the monitor screen. For the decoding of the monitor coordinates different phase unwrapping methods were presented. In addition, approaches were discussed on how the classical phase unwrapping methods can be improved. The main contribution of this chapter is a new probabilistic approach for phase unwrapping that uses circular statistics to describe the phase-shift coding. The presented method unwraps all phase measurements simultaneously by finding the coordinate that had the maximum probability to cause the phase measurements. Using circular statistics, both the periodicity of the phase is taken into account and the estimation of the phase uncertainty can be included in the unwrapping process, thus automatically compensating for individual erroneous phase measurements. This is achieved by expressing the individual phase measurements as appropriate stochastic variables, where different distributions were investigated to describe them. Using this, the probability density of the encoded coordinate could be determined, which allowed finding the optimal decoding by a maximum-likelihood approach. Thus, it became possible

to implicitly and simultaneously compensate for the wrapping of all phase measurements. Furthermore, it was demonstrated how to extend the presented probabilistic method to a spatio-temporal approach by integrating a local surface continuity assumption into the framework and modeling the local pixel neighborhood. This results in an implicit smoothing of the probability densities over the spatial dimensions. To ensure the assumptions are not violated, a modified edge detector is used to detect discontinuities in the surface and exclude them from the spatial modeling.

Simulations compared the presented methods with state-of-the-art temporal phase unwrapping algorithms and investigated the effect of different noise types. The results showed that the proposed probabilistic methods are noticeably more robust against noise. This provides the ability to increase the acquisition speed of the optical encoding by using phase-shift coding with fewer shifts, where the noise level is generally higher. It was also shown that the proposed methods allow a relatively free choice in the range of frequencies of the sinusoidal pattern sequence so that even rational frequencies yield good results. At the same time, it was demonstrated that by modeling the periodicity using circular probability densities, the unwrapping errors at the boundary of the coding interval can be significantly reduced. In addition, the inclusion of the phase uncertainty allows to automatically compensate for too noisy phase measurements, making the presented methods very robust against impulse noise. Although the *von Mises* distribution does not ideally describe the phase noise, it handles impulse distortions better than the model of Rathjen [167] and thus proves to be a suitable compromise to compensate well for both Gaussian noise and impulse noise at the same time. Because the image noise is in general unknown, model errors may be introduced. Nevertheless, the *von Mises* distribution again proved to be robust towards such errors. Finally, the extension of the temporal approach to a spatio-temporal approach can considerably increase the robustness of the method even further, eventually leading to improved accuracy of the camera-to-monitor registration. This provides ideal starting conditions for subsequent camera calibration procedures and the deflectometric reconstruction of specular surfaces.

5 System Calibration

In order to carry out a deflectometric measurement for specular surface reconstruction, it is not sufficient to measure only the simple imaging function as a registration between camera and monitor. With the registration, we may know a mapping of camera pixels to monitor coordinates, but the geometry of the scene cannot be reconstructed without knowing the exact geometry of the measurement setup as well. The setup must therefore be calibrated. To perform a triangulation measurement of the surface, an intrinsic and extrinsic calibration of both the camera and the reference monitor is necessary. The intrinsic calibration of the camera allows a calculation of the vision rays of the camera. Since light field cameras have a more complex optical structure than standard cameras and since deflectometry requires a highly accurate calibration, it is difficult to describe the light field camera sufficiently accurate using only a low-dimensional camera model. Therefore, the calibration of the light field camera in this thesis is done by adopting a generic camera model, in which the vision rays belonging to each pixel are estimated individually, thereby achieving a high precision calibration. As the main contribution of this chapter, an approach is presented that performs the generic calibration via an alternating optimization of the ray parameters and the unknown poses of a reference monitor. In addition, the positional uncertainty of the reference coordinates, which is obtained using the phase-shift coding, is taken into account in the optimization. In this context, the explicit intrinsic calibration of the monitor allows calculating the 3D coordinate of an observed monitor feature using the registration data. This improves the overall calibration result, due to possible deformations of the display being taken into account and the refraction on the front glass being compensated for. However, the coordinates are then still specified in the local coordinate system of the monitor. Only an extrinsic calibration of the whole measurement setup finally allows obtaining transformation parameters that connect the monitor coordinates and the

camera coordinates. In other words, the entire system calibration aims at providing a vision ray for each camera pixel and, additionally, it allows determining the 3D coordinates of a feature on the monitor in the camera coordinate system. Ultimately, this will then be used in Ch. 7 to calculate the surface normals of a specular object under examination.

In the following section, the basics of camera calibration will be explained using the classic pinhole camera model as an example. Subsequently, Sec. 5.2.1 introduces the generic camera model. Sec. 5.2 shows, how the model can be fitted to measurement data to estimate its parameters. In Sec. 5.3, the reference monitor is described, and it is shown how the monitor model and the estimation of its parameters can be integrated into the generic camera calibration. Finally, in Sec. 5.4, as the last part of the system calibration, the extrinsic calibration of the deflectometry measurement system is described. It returns the relative pose between camera and monitor. Sec. 5.5 concludes with an evaluation and analysis of the presented methods.

5.1 Principles of Camera Calibration

Probably the simplest and most widely used camera model is the pinhole camera model, see figure 5.1. It describes the projection of points in 3D space onto an image plane. The center of the projection is the origin of the camera coordinate system, and it is often referred to as the optical center. The image plane is located at a distance f from this center, and the line from the camera center perpendicular to the image plane is called the principal axis or optical axis. The point where this axis meets the image plane is called the principal point. In the pinhole camera model, a point in space with coordinates $\mathbf{x} = (x, y, z)^T$ is mapped to a point $(fx/z, fy/z, f)^T$ in the image plane [72]. Here, it is still assumed that the origin of the image coordinates in the image plane lies in the principal point, which is rarely the case for real cameras. Hence, a more general mapping from points in 3D space to points in image space is

$$(x, y, z)^T \rightarrow \left(\frac{fx}{z} + c_s, \frac{fy}{z} + c_t \right)^T, \quad (5.1)$$

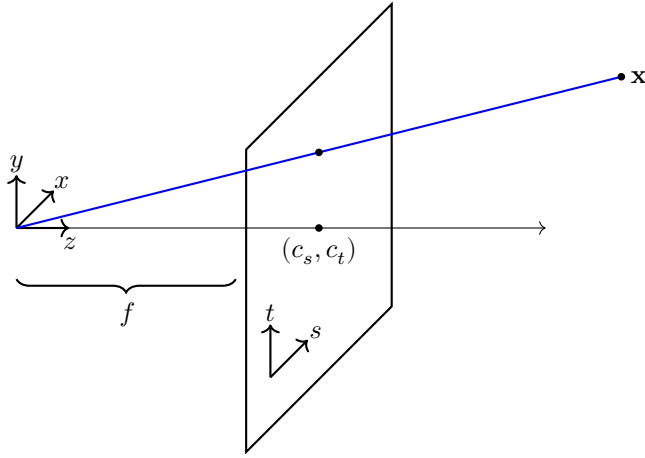


Figure 5.1 Pinhole camera model: A 3D point is perspectively projected onto an image plane that is placed at distance f to the origin.

where $(c_s, c_t)^T$ are the local 2D coordinates of the principal point in the image plane. If the world points are represented in homogeneous coordinates, the central projection can be expressed simply as matrix multiplication. And more generally, if a 3D point is first transformed into the camera coordinate system, the complete projection equation for the pinhole camera model is obtained [72]:

$$\lambda \begin{pmatrix} s \\ t \end{pmatrix} = \mathbf{K} (\mathbf{R} | \mathbf{t}) \begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix} = \begin{pmatrix} f_t & 0 & c_s \\ 0 & f_s & c_t \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}, \quad (5.2)$$

where s, t are the local coordinates in the image plane, and $\lambda = z$ is a scaling factor. The matrix \mathbf{K} represents the intrinsic parameters and is called the calibration matrix of the camera (or camera matrix for short). f_s and f_t represent the focal length of the camera on the s -axis and the t -axis, respectively. The coordinate transformation of the 3D point is described by the rotation matrix \mathbf{R} and the translation vector \mathbf{t} , which are the extrinsic parameters.

The pinhole model does not account for lens distortion. To accurately represent a real camera, radial and tangential lens distortion are often introduced. Radial distortion is when the magnitude of the aberration depends solely on the distance of the object point from the optical axis [16], and tangential distortions are caused by a lens that is not parallel to the image plane [79]. A convenient camera model can be derived by combining the pinhole model with a distortion correction [246]. *E.g.*, for the radial distortion, the observed image coordinates (\tilde{s}, \tilde{t}) can be calculated from the ideal coordinates (s, t) with

$$\tilde{s} = s + (k_1 r^2 + k_2 r^4 + k_3 r^6 + \dots) (s - c_s), \quad (5.3)$$

$$\tilde{t} = t + (k_1 r^2 + k_2 r^4 + k_3 r^6 + \dots) (t - c_t), \quad (5.4)$$

where $r^2 = (s - c_s)^2 + (t - c_t)^2$ and k_1, k_2, \dots are the distortion coefficients. Estimating all the intrinsic camera parameters $\mathbf{K}, k_1, k_2, \dots$ is then done by observing features (usually checkerboard features) on a reference target from different positions, and by minimizing the projection error [246]. This is the distance in pixels between the observed 2D positions of the 3D features on the image sensor and their projections to the sensor plane, which are calculated with the parametric camera model.

5.2 Generic Camera Calibration

Accurate optical measurement methods are becoming increasingly important for high-precision manufacturing. The rising demand can be satisfied by modern imaging systems with advanced optics. The exact geometric calibration of these systems is of essential importance for computer vision and optical metrology. Most systems use perspective projection with a single projection center and are referred to as central cameras. They can often be described by low-dimensional, parametric models with few intrinsic parameters, *e.g.*, the pinhole model from the previous section. In some applications in the field of optical metrology, more complex imaging systems are needed. These can often no longer be described by a central camera model and are in many cases non-parametric and non-central, *e.g.*, multi-camera systems, catadioptric cameras, or light field cameras [72, 137, 156, 195]. Here, more sophisticated models are needed, which always have to be precisely adapted to the specific camera.

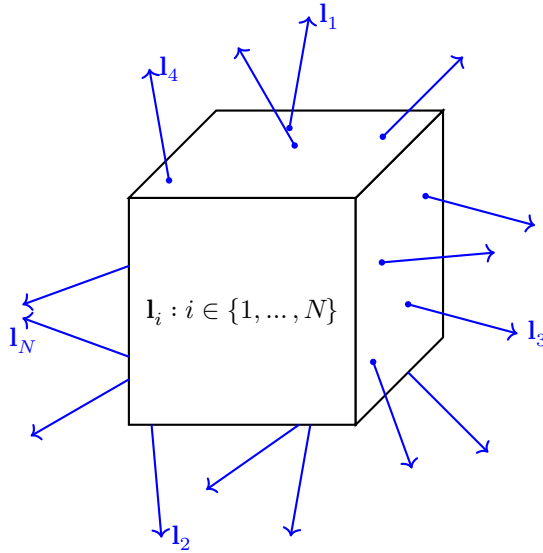


Figure 5.2 An imaging system guides light rays to photosensitive elements. The generic camera model characterizes the light rays outside the camera independently of the internal optics. Only a relation between the rays \mathbf{l}_i and the corresponding pixel index i is established.

5.2.1 The Generic Camera Model

The disadvantage of low-dimensional models is that they have poor descriptive power, and in modern cameras not every pixel of the many millions can be perfectly described by these models. The more complex an imaging system is, the more difficult it becomes to model it. The more elaborate the optical elements are, the more challenging it becomes to find a mathematically correct mapping between the light of the captured scene and the physical sensor plane of the camera. Consequently, in recent years, the lack of flexibility and precision has led to the development of new camera models, where cameras can be described as generic imaging systems, which are independent of the specific camera type and allow high-precision calibration.

The generic camera model was originally introduced in the works of Grossberg and Nayar [66, 67]. An arbitrary imaging system is modeled as

a non-parametric discrete black box containing photosensitive elements. Each pixel collects light from a bundle of rays that enter the imaging system, referred to as *raxel*, which consists of geometric ray coordinates and radiometric parameters. The set of all raxels builds the complete generic imaging model, see figure 5.2.

5.2.2 Related Works

The first work on the generic camera model was conducted by Grossberg and Nayar [66, 67]. The authors perform the calibration by measuring the intersection of camera rays with known reference targets: a monitor that is moved by a linear translation stage with known steps. To obtain the radiometric parameters, they control the intensity of light along the rays and measure the response in the image. Sturm and Ramalingam [191] and Ramalingam *et al.* [165] exclude the radiometric properties and propose a calibration of the generic model where the poses of the reference may be unknown. A closed-form solution can be obtained, if the same pixel sees three points of the reference object. The downside of their method is that the ray distribution of the camera has to be known in advance. For example, different models apply when the imaging system is non-central or a perspective camera, and complicated parametrization steps are necessary. Bothe *et al.* [23] and Miraldo *et al.* [131] achieve pixel-wise calibration by circumventing the estimation of the target pose by simply tracking it using an external stereo camera system or an IR tracker, respectively. Bergamasco *et al.* [15], on the other hand, assume unknown poses and calibrate the camera by iteratively calculating the projection of the rays onto a coded calibration monitor, and by minimizing the resulting coding error on a pixel level. In addition, they estimate the reference pose using an adapted iterative closest point method. Miraldo and Araujo [130] reduce the number of parameters by fitting a spline surface onto the set of rays. Thus, they evaluate the camera on a subset of control points. Rosebrock [171] additionally includes the measurement uncertainty of the reference target into the calibration procedure by iteratively updating this spline surface. However, these spline-based methods only work when the imaging system is smooth, *i.e.*, multi-camera systems, light field cameras, or other more complex optical systems are excluded and cannot be modeled using this approach.

Apart from calibration, the generic camera model is used in the field of pose estimation, structure from motion, and surface reconstruction. Guo *et al.* [68] calibrate a generic camera system using a linear translation stage and then aim to estimate the pose of a target object by orthographically projecting it onto the calibration planes to approximate the true object pose in an iterative manner. Kneip and Furgale [104] propose UPnP that generalizes the absolute pose problem to general cameras by finding a closed-form least-squares solution for the absolute pose. Lee *et al.* [108] mount a multi-fish-eye camera system on a robotic car platform and use the generic camera model to track the position while driving. Albarelli *et al.* [5] use the generic imaging model in a structured light 3D scanning system, where they use a generic model for both the camera and the projector.

5.2.3 Alternating Minimization-Based Calibration

The goal of the following sections is to find a flexible calibration procedure that can accurately describe the geometric properties of an arbitrary imaging system using the generic camera model. In the end, however, one does not obtain an “image”, but rather a set of rays with corresponding intensities. Still, this does not interfere with many applications in optical metrology, *e.g.*, laser triangulation, profilometry, or deflectometry, where mostly the geometric ray properties are relevant [11, 145, 209]. The presented method assumes unknown poses of the calibration target and iteratively solves the subproblems of camera calibration and pose estimation without the use of an additional translation or rotation stage. By processing every pixel individually and updating each pose one at a time, the computational costs can efficiently be reduced, whereby every camera ray and each observed reference point contribute to the result.

The portion of the light that is sampled by a single pixel has a cone-shaped expansion due to the effects of the depth-of-field. For simplicity, this work models a raxel as a ray running through the center of this cone along the direction of light propagation. There are various possibilities for a mathematical description of rays, yet in this work, the concept of Plücker-coordinates as described in Sec. 2.3 is used. In 6D-Plücker-space a Plücker-line $\mathbf{l} = (\mathbf{d}^T, \mathbf{m}^T)^T \in \mathbb{P}^6$ is defined by its direction vector $\mathbf{d} \in \mathbb{R}^3$ and its moment vector $\mathbf{m} \in \mathbb{R}^3$ with the constraints $\|\mathbf{d}\| = 1$, $\mathbf{d}^T \mathbf{m} = 0$.

Calibrating the geometric properties of a camera using a generic camera model means that for each individual pixel its ray \mathbf{l} , with the direction vector \mathbf{d} and the moment vector \mathbf{m} , must be estimated. This can be done in the usual way: estimating all unknown parameters by minimizing an error function. In the traditional camera calibration approach, one has to minimize the projection error, as a distance between the projection of an observed target feature onto the sensor plane and the observing pixel of the same feature. However, due to the independence of the rays from the actual physical camera system, when considering a generic model, such an error measure cannot be used, because there does not exist a model for the sensor plane. As an alternative, the ray re-projection error should be minimized instead, which represents the distance between the ray and the observed feature in 3D space. In conclusion, ray parameters are sought that minimize a suitable distance measure between the camera rays and observed reference points, whereby the positions of the references in the local coordinate system are assumed to be unknown. Figure 5.3 illustrates the approach. The calibration can now be formulated in the sense of a least-squares problem by minimizing

$$f(\mathcal{R}, \mathcal{T}, \mathcal{L}) = \sum_{k,i} d(\mathbf{p}_{ik}, \mathbf{l}_i)^2. \quad (5.5)$$

Here, the index i represents the individual rays and k depicts the index of the reference target pose. The metric $d(\cdot, \cdot)$ is a suitable ray-to-point distance and $\mathbf{p}_{ik} = \mathbf{R}_k \mathbf{x}_{ik} + \mathbf{t}_k$ are the observed features in 3D space, where \mathbf{x}_{ik} is a local point on a reference target. The matrix $\mathbf{R}_k \in \text{SO}(3)$ and the vector $\mathbf{t}_k \in \mathbb{R}^3$ are the corresponding transformations to the camera coordinate system. And for a compact notation, the set of rotations, translations and rays are defined to be

$$\mathcal{R} := \{\mathbf{R}_1, \mathbf{R}_2, \mathbf{R}_3, \dots\}, \quad (5.6)$$

$$\mathcal{T} := \{\mathbf{t}_1, \mathbf{t}_2, \mathbf{t}_3, \dots\}, \quad (5.7)$$

$$\mathcal{L} := \{\mathbf{l}_1, \mathbf{l}_2, \mathbf{l}_3, \dots\}. \quad (5.8)$$

To present the camera calibration as a per-pixel problem and to treat each pixel independently of its neighbors, sufficient observations of reference features have to be available for every pixel. However, the widely used checkerboard patterns can provide only sparse features which are

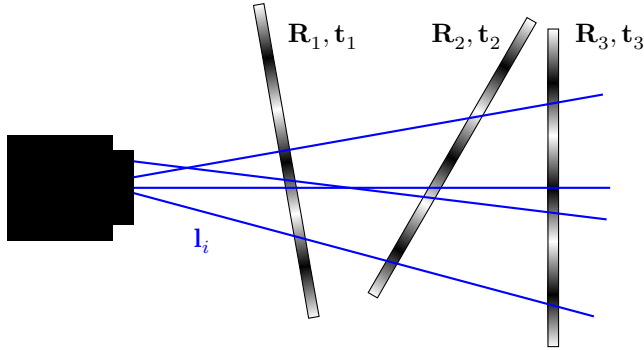


Figure 5.3 Generic calibration: The imaging system is treated as a black box that is independent of the internal optics described by a set of vision rays l_i . Each individual ray observes the intersected reference target point. The ideal calibration results in a minimal distance between rays and observed reference feature points.

not nearly enough for generic camera calibration. Instead, it is a good idea to use active targets, *e.g.*, flat monitor displays, and active encoding strategies, to assign each camera ray a 2D point in the local reference target plane. Thus, each ray can observe one feature per pose. In this work, the detection of features in the reference target plane and with it the registration of camera rays l_i to monitor display points x_{ik} is found via a temporal coding of the monitor pixels. Hence, the presented multi-frequency phase-shift coding with the proposed probabilistic phase unwrapping from Ch. 4 can be used to obtain highly accurate reference features and their respective point uncertainties. Of course, to use the spatio-temporal phase unwrapping, the mapping of reference features onto camera pixels needs to be sufficiently smooth. That is, the reference target has to have a smooth surface, and in addition, the mapping of camera pixels onto the corresponding ray surface needs to be a continuous function. When standard cameras are used, this smoothness assumption can be easily met. Though, for more complex camera systems, problems may arise. In particular, the MLA-based light field cameras that are investigated in this work show a strong discontinuous behavior near the edges of the microlenses, hence violating the smoothness assumption. Nonetheless,

these areas can easily be detected with the edge detection presented in Sec. 4.4.4, and can thus be excluded from the spatial modeling.

With the previous results, an objective function can be defined that needs to be minimized to calibrate the camera and find all ray parameters $\mathbf{d}_i, \mathbf{m}_i$. Simultaneously, the estimation of the pose of the calibration targets $\mathbf{R}_k, \mathbf{t}_k$ with respect to the camera is performed. This is done in a weighted-least-squares sense by minimizing the distance between uncertain target points \mathbf{x}_{ik} , whose uncertainty is $\sigma_{ik}^2 := \sigma_{x_{ik}}^2 + \sigma_{y_{ik}}^2$ and their corresponding camera rays. To this end, the phase-shift coding strategy is utilized to estimate the uncertainties of the reference target points, which results in a weighting factor $w_{ik} = \sigma_{ik}^{-2}$. In conclusion, the objective function for the generic camera calibration is obtained:

$$f(\mathcal{R}, \mathcal{T}, \mathcal{L}) = \sum_{i,k} w_{ik} \|(\mathbf{R}_k \mathbf{x}_{ik} + \mathbf{t}_k) \times \mathbf{d}_i - \mathbf{m}_i\|^2. \quad (5.9)$$

Regardless of the used distance measure, it is very difficult to minimize such a problem in a reasonable time and with the appropriate use of computational resources. The ray model with six parameters and two constraints has four degrees of freedom per pixel. Especially for today's standard cameras, this leads to a huge number of ray parameters that have to be optimized, *e.g.*, a 40-megapixel camera has 240 million parameters. In addition, the reference target pose is in general not known. This means that at the same time six degrees of freedom per pose have to be estimated. The coupling of poses and rays and the immense number of parameters result in an extremely high-dimensional problem that cannot be solved using a single optimization method. The calculation of a gradient or a Hessian and the corresponding function evaluations would be computationally too expensive.

Therefore, it is useful to divide the problem into subproblems and then solve them iteratively in the sense of an *Alternating Minimization* (AM) [70, 139]. Accordingly, problem (5.9) is split into a camera calibration and a reference target pose estimation. The approach of an AM is to fix a

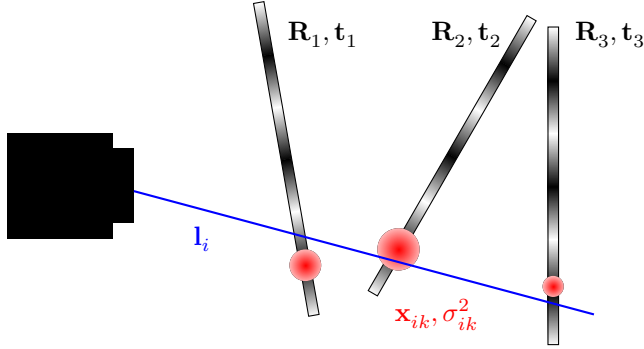


Figure 5.4 Generic ray estimation: Each individual ray observes several uncertain reference features. The optimal ray has a minimal distance to all observed points.

parameter set and solve the resulting problem. This way, one has two particular problems to solve in each iteration:

$$\mathbf{l}_i^{(n)} = \arg \min_{\mathbf{l}_i \in \mathbb{P}^6} f(\mathcal{R}^{(n-1)}, \mathcal{T}^{(n-1)}, \mathbf{l}_i) \quad \forall i = 1, \dots, I, \quad (5.10)$$

$$\mathbf{R}_k^{(n)}, \mathbf{t}_k^{(n)} = \arg \min_{(\mathbf{R}_k, \mathbf{t}_k) \in \text{SE}(3)} f(\mathbf{R}_k, \mathbf{t}_k, \mathcal{L}^{(n)}) \quad \forall k = 1, \dots, K, \quad (5.11)$$

where an appropriate initialization $\mathcal{R}^{(0)}, \mathcal{T}^{(0)}$ has to be chosen. The first optimization problem is solved for each pixel i individually by fixing all the reference target poses and the second problem is solved for every single pose k by assuming fixed ray parameters. This allows for solving the subproblems more easily. It will be shown that optimal solutions can be found in each iteration, which further leads to the overall alternating minimization converging towards a solution.

5.2.4 Generic Ray Estimation

One step in the camera calibration procedure is to estimate the ray parameters by assuming known poses of the calibration targets. This greatly reduces the complexity. Instead of calculating every parameter at once, one can calibrate the ray $\mathbf{l}_i = (\mathbf{d}_i^T, \mathbf{m}_i^T)^T \in \mathcal{L}$ of each pixel individually

(or in parallel). Hence, for every single ray, a separate optimization problem is obtained, as illustrated in figure 5.4. To simplify the optimization, the objective function is written in the more compact form

$$\begin{aligned}
 f(\mathbf{d}_i, \mathbf{m}_i) &= \sum_k w_{ik} \|\mathbf{p}_{ik} \times \mathbf{d}_i - \mathbf{m}_i\|^2 \\
 &= \sum_k w_{ik} ([\mathbf{p}_{ik}]_{\times} \mathbf{d}_i - \mathbf{m}_i)^T ([\mathbf{p}_{ik}]_{\times} \mathbf{d}_i - \mathbf{m}_i) \\
 &= \sum_k w_{ik} (\mathbf{d}_i^T [\mathbf{p}_{ik}]_{\times}^T [\mathbf{p}_{ik}]_{\times} \mathbf{d}_i + \mathbf{m}_i^T 2 [\mathbf{p}_{ik}]_{\times}^T \mathbf{d}_i + \|\mathbf{m}_i\|^2) \\
 &= \mathbf{d}_i^T \mathbf{A}_{\text{dd},i} \mathbf{d}_i + \mathbf{m}_i^T \mathbf{A}_{\text{md},i} \mathbf{d}_i + a_{\text{mm},i} \|\mathbf{m}_i\|^2, \quad (5.12)
 \end{aligned}$$

where the vector $\mathbf{p}_{ik} = \mathbf{R}_k \mathbf{x}_{ik} + \mathbf{t}_k$ represents the reference target points in camera coordinates. In addition, for better readability, the index i is neglected in the remainder of this section.

Since \mathbf{A}_{dd} is derived from a sum of products of two mutually transposed matrices, it is always positive semidefinite. In addition, it is invertible as long as at least two different points \mathbf{p}_{ik} are observed. Thus, problem (5.12) is convex. Considering the characteristics of the Plücker-rays (2.14), finding the optimal rays results in minimizing a quadratic program with quadratic equality constraints: $\mathbf{d}^T \mathbf{m} = 0$, $\|\mathbf{d}\| = 1$. Although the minimization of such a problem, in general, requires a difficult nonlinear minimization, the following presents a solution to find a global minimum in this specific case, using a few simple steps.

At first, it should be obvious that the solution of the constraint problem is scale ambiguous and that the norm of the ray direction $\|\mathbf{d}\|$ does not influence the actual ray properties [207]. Thus, after having found a solution, applying a normalization to the ray $\mathbf{l}_n = \mathbf{l} / \|\mathbf{d}\| = (\mathbf{d} / \|\mathbf{d}\|, \mathbf{m} / \|\mathbf{d}\|)$ makes it possible to obtain a geometrical meaningful point-to-ray distance (2.20). To deal with the equality constraints, making use of the method of Lagrange multipliers helps. Hence, the constraints are added to the objective function using the Lagrange multipliers λ, μ :

$$g = \mathbf{d}^T \mathbf{A}_{\text{dd}} \mathbf{d} + \mathbf{m}^T \mathbf{A}_{\text{md}} \mathbf{d} + a_{\text{mm}} \|\mathbf{m}\|^2 + \lambda \mathbf{d}^T \mathbf{m} + \mu (\mathbf{d}^T \mathbf{d} - 1). \quad (5.13)$$

Further, stationary points of this Lagrangian can be found by fulfilling the first order conditions for a minimum:

$$\partial_{\mathbf{d}}g = 2\mathbf{A}_{\text{dd}}\mathbf{d} + \mathbf{A}_{\text{md}}^T\mathbf{m} + \lambda\mathbf{m} + 2\mu\mathbf{d} \stackrel{!}{=} \mathbf{0}, \quad (5.14)$$

$$\partial_{\mathbf{m}}g = 2a_{\text{mm}}\mathbf{m} + \mathbf{A}_{\text{md}}\mathbf{d} + \lambda\mathbf{d} \stackrel{!}{=} \mathbf{0}, \quad (5.15)$$

$$\partial_{\lambda}g = \mathbf{d}^T\mathbf{m} \stackrel{!}{=} 0, \quad (5.16)$$

$$\partial_{\mu}g = \|\mathbf{d}\|^2 - 1 \stackrel{!}{=} 0. \quad (5.17)$$

Using (5.15) and (5.16) results in the solution for the ray moment \mathbf{m} and the multiplier λ :

$$\mathbf{m} = -\frac{1}{2a_{\text{mm}}}(\mathbf{A}_{\text{md}} + \lambda\mathbf{I})\mathbf{d}, \quad (5.18)$$

$$\mathbf{d}^T\mathbf{m} = -\frac{1}{2a_{\text{mm}}}\mathbf{d}^T(\mathbf{A}_{\text{md}} + \lambda\mathbf{I})\mathbf{d} \stackrel{!}{=} 0, \quad (5.19)$$

$$\begin{aligned} \Rightarrow \lambda &= -\frac{\mathbf{d}^T\mathbf{A}_{\text{md}}\mathbf{d}}{\mathbf{d}^T\mathbf{d}} \stackrel{(5.17)}{=} -\mathbf{d}^T\mathbf{A}_{\text{md}}\mathbf{d} = -\mathbf{d}^T\left(\sum_k 2w_{ik}[\mathbf{p}_{ik}]_{\times}^T\right)\mathbf{d} \\ &= -\mathbf{d}^T\left(\left(\sum_k 2w_{ik}\mathbf{p}_{ik}\right) \times \mathbf{d}\right) = -\mathbf{d}^T(\mathbf{p} \times \mathbf{d}) = 0, \end{aligned} \quad (5.20)$$

where the last equation holds because \mathbf{d} is orthogonal to $\mathbf{p} \times \mathbf{d}$, $\forall \mathbf{p} \in \mathbb{R}^3$. Inserting these results into (5.14) leads to a simple eigenvalue problem for the solution of the ray direction \mathbf{d} and the Lagrange multiplier μ :

$$\left(\mathbf{A}_{\text{dd}} - \frac{1}{4a_{\text{mm}}}\mathbf{A}_{\text{md}}^T\mathbf{A}_{\text{md}}\right)\mathbf{d} = -\mu\mathbf{d}. \quad (5.21)$$

This equation still contains the trivial solution $\mathbf{d} = \mathbf{m} = \mathbf{0}$ which however has no geometric meaning for the calibration and is excluded by (5.17). Apart from that, the solution space of (5.21) consists of three eigenvalues μ_j with corresponding eigenvectors \mathbf{d}_j . After estimating a possible \mathbf{d}_j and corresponding Lagrange multiplier μ_j , it is necessary to scale the eigenvalue problem in order to normalize the ray such that $\|\mathbf{d}_j\| = 1$. This preserves the geometric meaning of (2.20) and allows obtaining an unambiguous scaling. Further, (5.18) provides the corresponding ray momentum \mathbf{m}_j . And finally, from these at most three possible stationary points, the one with the smallest objective function

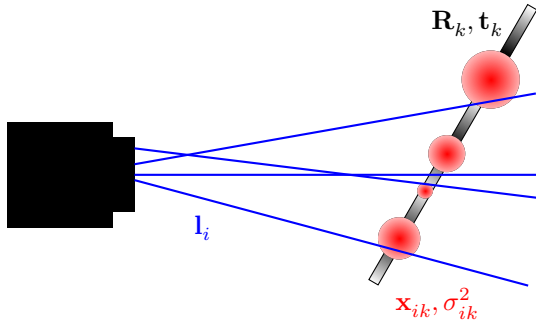


Figure 5.5 Generic pose estimation: The set of all rays observe features on the calibration reference. The optimal pose estimation results in a minimal distance between the rays and corresponding feature points.

value (5.13) is selected to be the optimal solution. In conclusion, one finds a closed-form solution for the least-squares problem of the weighted ray-to-point distance minimization.

5.2.5 Generic Pose Estimation

As before, the estimation of the calibration target pose can drastically be simplified by assuming known ray parameters. Therefore, it becomes possible to optimize each pose individually, as illustrated in figure 5.5. The objective function for each pose k becomes:

$$f(\mathbf{R}_k, \mathbf{t}_k) = \sum_i w_{ik} \|(\mathbf{R}_k \mathbf{x}_{ik} + \mathbf{t}_k) \times \mathbf{d}_i - \mathbf{m}_i\|^2. \quad (5.22)$$

However, solving for a pose $\mathbf{R}_k, \mathbf{t}_k$ is non-trivial because the solution space is restricted to the special Euclidean group $\text{SE}(3)$, which combines rotations and translations in three dimensions, $\mathbf{R}_k \in \text{SO}(3)$ and $\mathbf{t}_k \in \mathbb{R}^3$, respectively. Directly applying a nonlinear optimization procedure is not advisable, because every function evaluation results in the summation over all rays and is thus computationally very expensive. Therefore, as before, a more compact form of this quadratic function is necessary to reduce the computational effort.

Again for the sake of brevity, the index k is omitted for the remainder of this section. For further simplification, the vectorization operator $\mathbf{r} = \text{vec}(\mathbf{R}) \in \mathbb{R}^9$ stacks the columns of the 3×3 matrix \mathbf{R} . By computing the summation over all ray indices i only once, reordering, and extracting the pose parameters, the objective function can be formulated independently of the actual number of rays, which simplifies and speeds up the following optimization steps (see Appx. 9.1.1 for details):

$$f(\mathbf{R}, \mathbf{t}) = \mathbf{r}^T \mathbf{A}_{\text{rr}} \mathbf{r} + \mathbf{t}^T \mathbf{A}_{\text{tt}} \mathbf{t} + \mathbf{t}^T \mathbf{A}_{\text{tr}} \mathbf{r} + \mathbf{b}_{\text{r}}^T \mathbf{r} + \mathbf{b}_{\text{t}}^T \mathbf{t} + h \quad (5.23)$$

subject to $\mathbf{r} = \text{vec}(\mathbf{R})$, $(\mathbf{R}, \mathbf{t}) \in \text{SE}(3)$.

While observing the constraint quadratic objective (5.23), one may notice that the main constraint lies in the rotational part and the objective is also convex in the translational part. Thus, the problem can further be reduced by decoupling translation and rotation, which means that \mathbf{t} can be expressed in terms of \mathbf{R} . The first order condition for a minimum $\partial_{\mathbf{t}} f(\mathbf{R}, \mathbf{t}) = 0$ leads to the optimal translation vector

$$\mathbf{t} = -\frac{1}{2} \mathbf{A}_{\text{tt}}^{-1} (\mathbf{A}_{\text{tr}} \mathbf{r} + \mathbf{b}_{\text{t}}). \quad (5.24)$$

Inserting (5.24) into (5.23) results in the decoupling of the rotation and translation subproblem, which then again yields a new quadratic optimization problem (see Appx. 9.1.1):

$$f(\mathbf{R}) = \mathbf{r}^T \mathbf{A}_{\text{r}} \mathbf{r} + \mathbf{b}^T \mathbf{r} + c, \quad \text{subject to } \mathbf{r} = \text{vec}(\mathbf{R}), \mathbf{R} \in \text{SO}(3). \quad (5.25)$$

After finding a solution for the rotation matrix, the optimal translation vector is derived from (5.24), assuming invertibility of \mathbf{A}_{tt} . As shown in Appx. 9.1.3, the matrix \mathbf{A}_{tt} is positive definite in most cases, except for a few exotic camera ray distributions, *e.g.*, parallel rays, telecentric optics. Hence, the equation truly finds the minimum of the objective with respect to the translation.

Although minimization of (5.25) seems simple at first, the optimization has the constraint to find a solution in $\text{SO}(3)$. This is equivalent to a non-convex problem with quadratic and cubic constraints on the rotation parameters, *cf.* Sec. 2.2. For solving this, there exist various approaches in the literature. Bergamasco *et al.* [15] use an iterative closest

point algorithm that iteratively calculates the transformation from the observed points to the closest point on the corresponding rays, which however only converges near the optimum. Kanatani [101] suggests a fast method by first calculating a Euclidean solution by first assuming $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ and then projecting the solution onto the $\text{SO}(3)$ -manifold using the singular value decomposition, which results in a not entirely correct minimization.

However, since the main focus of this work is not real-time optimization, but rather a highly precise pose estimation, there is the obligation to find an accurate minimum to ensure convergence of the AM calibration. Therefore, a gradient-based optimization approach on the Riemannian manifold $\text{SO}(3)$ with tangent space $\mathfrak{so}(3)$ is applied, *cf.* Sec. 2.2. The tangent space to the Lie group $\text{SO}(3)$ is its Lie algebra $\mathfrak{so}(3)$, which consists of all skew-symmetric 3×3 matrices. The mapping from any element $[\xi]_{\times} \in \mathfrak{so}(3)$ to $\mathbf{R} \in \text{SO}(3)$ is called the exponential map $\mathbf{R} = \text{Exp}([\xi]_{\times}) = e^{[\xi]_{\times}}$, and the reverse map is called the logarithmic map $[\xi]_{\times} = \text{Log}(\mathbf{R})$. Both can be calculated in closed form using the well known Rodrigues rotation formulas (2.12), (2.13). Therefore, in a local neighborhood $g_{\mathbf{R}}(\xi) = \text{Exp}([\xi]_{\times}) \mathbf{R}$ one can find a parametrization of the manifold in the tangent space. A function defined on the manifold can thus be described locally by Euclidean coordinates $\xi \in \mathbb{R}^3$:

$$\begin{aligned} f \circ g \circ [\cdot]_{\times} : \mathbb{R}^3 &\rightarrow \mathfrak{so}(3) \rightarrow \text{SO}(3) \rightarrow \mathbb{R}, \\ f_{\xi}(\mathbf{R}) &:= f(g_{\mathbf{R}}(\xi)) = f(\text{Exp}([\xi]_{\times}) \mathbf{R}). \end{aligned} \quad (5.26)$$

If a function is to be optimized on the manifold, the corresponding direction of descent must be sought in the local tangent space $f_{\xi}(\mathbf{R})$. To use conventional optimization methods, a valid representation for both the gradient and the Hessian must be identified. According to Absil *et al.* [1], these can be easily found by using directional derivatives of the locally parameterized manifold in the direction of the tangent space:

$$D_{\xi} f(\mathbf{R}) = \partial_{\varepsilon} f_{\varepsilon \xi}(\mathbf{R}) \big|_{\varepsilon=0} = \xi^T \text{grad}(f), \quad (5.27)$$

$$D_{\xi} \text{grad}(f) = \partial_{\varepsilon^2}^2 f_{\varepsilon \xi}(\mathbf{R}) \big|_{\varepsilon=0} = \xi^T \text{Hess}(f) \xi. \quad (5.28)$$

Looking back at the original problem (5.25), this approach leads to the explicit formulas for the Riemannian gradient and Riemannian Hessian (see Appx. 9.1.2 for a detailed derivation of the operators):

$$\text{grad}(f) = 2\mathbf{Z}^T (\mathbf{R} \otimes \mathbf{I}) (\mathbf{A}\mathbf{r} + \mathbf{b}) , \quad (5.29)$$

$$\text{Hess}(f) = 2\mathbf{Z}^T ((\mathbf{R} \otimes \mathbf{I}) \mathbf{A} (\mathbf{R} \otimes \mathbf{I})^T - \mathbf{I} \otimes \text{mat}(\mathbf{A}\mathbf{r} + \mathbf{b}) \mathbf{R}^T) \mathbf{Z} , \quad (5.30)$$

with $\mathbf{Z} = [\text{vec}([\mathbf{e}_1]_{\times}), \text{vec}([\mathbf{e}_2]_{\times}), \text{vec}([\mathbf{e}_3]_{\times})] \in \mathbb{R}^{9 \times 3}$, the unit base vectors $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$, and the identity matrix \mathbf{I} . The reshape operator $\text{mat}(\cdot)$ is the inverse of the vectorization operator $\text{vec}(\cdot)$, and \otimes represents the Kronecker product.

After the formulas for the gradient and the Hessian have been established, a quadratic model of the local tangent space then enables to minimize the objective (5.25) with the help of an appropriate Newton descend algorithm. Apart from minor differences, the procedure is quite similar to the classic Euclidean approach [24]. For the current iteration, $\text{grad}f(\mathbf{R}^{(n)})$ and $\text{Hess}f(\mathbf{R}^{(n)})$ are calculated. After the search direction $\boldsymbol{\xi}^{(n)}$ has been found by solving the Newton equation, one has to calculate a projection of the tangent space back to the manifold to obtain a valid descend:

$$\text{Hess}f(\mathbf{R}^{(n)}) \boldsymbol{\xi}^{(n)} = -\text{grad}f(\mathbf{R}^{(n)}) , \quad (5.31)$$

$$\mathbf{R}^{(n+1)} = \text{Exp}(\alpha [\boldsymbol{\xi}^{(n)}]_{\times}) \mathbf{R}^{(n)} . \quad (5.32)$$

Finally, a subsequent 1D backtracking line search in $\text{SO}(3)$ finds a sufficient step size α and accelerates the convergence [140]. Figure 5.6 visualizes the procedure. In order to initialize the algorithm, an appropriate start is required, where in the context of an AM-camera-calibration, the pose estimate from the previous iteration may be used.

Looking back at the original camera pose optimization (5.23), we see that the pose has to be found in the special Euclidean group $\text{SE}(3)$. Optimization on this manifold is not straightforward, but the problem can be simplified by making use of the local diffeomorphism between the manifolds $\text{SE}(3)$ and $\text{SO}(3) \times \mathbb{R}^3$. If there is a (local) minimum in $\text{SE}(3)$, then the same minimum exists in $\text{SO}(3) \times \mathbb{R}^3$ [189]. Having this in mind, the presented optimization performs two steps: first optimization in $\text{SO}(3)$, using the manifold Newton descend; and afterward optimization in \mathbb{R}^3 , using (5.24). Performing the optimization in this manner might

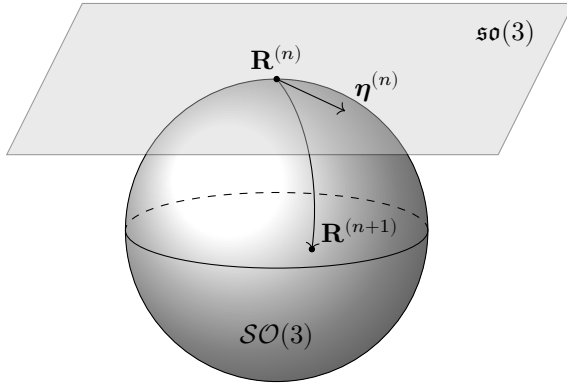


Figure 5.6 Local parametrization of $SO(3)$ -manifold through its tangent space $\mathfrak{so}(3)$. The search direction is found in the tangent space and projected back onto the manifold to find a minimum.

be less efficient in terms of iterations, but it yields the same optimization result while avoiding the more complex calculation in the $\mathfrak{se}(3)$ tangent space, which has a greatly different exponential and logarithmic map.

5.2.6 Convergence, Acceleration and Initialization

Depending on the current pose estimation, the camera ray calibration provides the globally optimal solution in every step. Furthermore, the pose estimation converges towards a minimum and provides no inferior result than the previous iteration. Following the research in the field of AM [65, 70], it is easy to show the convergence of the optimization procedure to a stationary point with an $\mathcal{O}(1/n)$ convergence rate. To obtain a faster convergence, acceleration techniques may be applied. Therefore, Nesterov's acceleration scheme is modified to obtain an almost $\mathcal{O}(1/n^2)$ convergence rate [62, 134]. The basic principle of this acceleration is that the difference between the new estimate and the old estimate is weighted and added to the new estimate in each iteration, where the weighting factor is a monotonically increasing sequence. However, these algorithms cannot be applied to the manifold optimization problems presented here without any adaptation. Hence, during the acceleration step, a weighted rate of the change of the pose parameters is added to the next estimate.

Algorithm 2 Accelerated Alternating Minimization

Input: For every pixel i and target pose k : measure monitor coordinates \mathbf{x}_{ik} and weight w_{ik}

Output: Calibrated ray \mathbf{l}_i for each pixel and pose $\mathbf{R}_k, \mathbf{t}_k$ of all references

Initialize: Set poses of reference targets $\mathcal{R}^{(0)}, \mathcal{T}^{(0)}$, set acceleration parameter $\alpha_0 = 1$

```

1: for  $n = 1, 2, 3, \dots$  do
2:   for  $i = 1, 2, 3, \dots$  do
3:     Hold pose parameters and optimize rays
4:      $\mathbf{l}_i^{(n+1)} = \arg \min_{\mathbf{l}_i \in \mathbb{P}^6} f(\mathcal{R}^{(n)}, \mathcal{T}^{(n)}, \mathbf{l}_i)$ 
5:   end for
6:   for  $k = 1, 2, 3, \dots$  do
7:     Hold ray parameters and optimize poses
8:      $\mathbf{R}_k^*, \mathbf{t}_k^* = \arg \min_{(\mathbf{R}_k, \mathbf{t}_k) \in \text{SE}(3)} f(\mathbf{R}_k, \mathbf{t}_k, \mathcal{L}^{(n+1)})$ 
9:     Update acceleration rate
10:     $\alpha_n = \frac{1 + \sqrt{4\alpha_{n-1}^2 + 1}}{2}$ 
11:    Accelerate translation and rotation update
12:     $\mathbf{t}_k^{(n+1)} = \mathbf{t}_k^* + \frac{\alpha_{n-1} - 1}{\alpha_n} (\mathbf{t}_k^* - \mathbf{t}_k^{(n)})$ 
13:     $\mathbf{R}_k^{(n+1)} = \text{Exp}\left(\frac{\alpha_{n-1} - 1}{\alpha_n} \text{Log}(\mathbf{R}_k^{\text{T}(n)} \mathbf{R}_k^*)\right) \mathbf{R}_k^*$ 
14:   end for
15: end for

```

When accelerating the rotation, of course, this has to be done on the $\text{SO}(3)$ -manifold: The current rotation is reversed by the previous rotation, projected onto the $\mathfrak{so}(3)$ tangent space using the Log-map, weighted by an acceleration parameter, and finally transformed back into a rotation matrix using the Exp-map and multiplied onto the current estimate. Algorithm 2 summarizes the complete accelerated AM calibration.

Although this is a strictly convergent algorithm, obviously no unique solution exists. Depending on the initialization, the optimization runs into an arbitrary coordinate system. Therefore, it is advisable to initialize the algorithm with a rough estimate of the reference target poses,

which could for example be obtained using standard model-based approaches presented in the literature [26, 246] or the generic approach by Ramalingam *et al.* [165]. However, here it is of utmost importance that the camera model is properly chosen. Alternatively, of course, one can also randomly select starting poses with the downside of a longer optimization time and the increased risk to converge to a non-optimal local minimum. Nonetheless, the arbitrary coordinate system poses no problem, since it does not change the geometric properties of the rays, and accordingly, the calibrated camera can be used without loss of accuracy. Even more, the final calibration can be easily transformed into a standardized coordinate system.

5.2.7 Normalizing the Ray Bundle

Due to the black box character of the generic calibration, it is initially not possible to define a consistent camera coordinate system for every calibrated camera. Even when using the same calibration algorithm for the same camera, the outcome can vary. Hence, the result of a generic calibration is in general not unique. That is, the calibrated camera rays are represented in an arbitrary coordinate system, which usually depends on the starting configuration of the generic calibration procedure or the used calibration reference target. Therefore, to transform this arbitrary coordinate system into one that is fixed to the individual camera, a few steps are necessary.

First, the origin of the camera coordinate system is defined to be the optical center of the camera. For central cameras or nearly-central cameras, *e.g.*, light field cameras, this corresponds approximately to the center of the exit pupil. Its location can be understood as the point \mathbf{p}_o that has the smallest distance to all rays, *i.e.*, it can be calculated by minimizing the weighted mean of the Euclidean distances to all rays:

$$\mathbf{p}_o = \arg \min_{\mathbf{p}} \sum_i w_i \|\mathbf{p} \times \mathbf{d}_i - \mathbf{m}_i\|^2. \quad (5.33)$$

The weighting factor w_i can be chosen to suppress poorly calibrated rays and to remove outliers. For instance, a simple choice is to use the inverse of the mean ray re-projection error

$$\varepsilon_i := \sum_k w_{ik} \|\mathbf{p}_{ik} \times \mathbf{d}_i - \mathbf{m}_i\|^2 \quad (5.34)$$

that can be calculated during the generic calibration procedure for each ray. This results in

$$\mathbf{p}_o = \left(\sum_i w_i [\mathbf{d}_i]_{\times} [\mathbf{d}_i]_{\times}^T \right)^{-1} \sum_i w_i [\mathbf{d}_i]_{\times} \mathbf{m}_i. \quad (5.35)$$

As a next step, the z -axis of the camera-fixed coordinate system defines the view axis as the average ray direction which can be found by solving the constrained optimization problem

$$\mathbf{d}_z = \arg \max_{\mathbf{d}} \sum_i w_i \langle \mathbf{d}, \mathbf{d}_i \rangle^2, \text{ subject to } \|\mathbf{d}\| = 1. \quad (5.36)$$

Using the Lagrange multiplier formalism and solving for \mathbf{d} produces an eigenvalue problem:

$$\mathbf{d}_z = \arg \max_{\mathbf{d}} \sum_i w_i \langle \mathbf{d}, \mathbf{d}_i \rangle^2 - \mu (\mathbf{d}^T \mathbf{d}), \quad (5.37)$$

$$\Rightarrow \left(\sum_i w_i \mathbf{d}_i \mathbf{d}_i^T \right) \mathbf{d}_z = \mu \mathbf{d}_z, \quad (5.38)$$

where the eigenvector \mathbf{d}_z with largest absolute eigenvalue μ results in the average ray direction. A corresponding rotation matrix, which rotates the bundle of rays from the old z -axis \mathbf{e}_z into the new z -direction, can then directly be calculated using the Rodrigues formula (2.12):

$$\mathbf{R}_z = \text{Exp}(\arccos(\mathbf{d}_z^T \mathbf{e}_z) (\mathbf{d}_z \times \mathbf{e}_z)). \quad (5.39)$$

The last remaining degree of freedom is the rotation around this new z -axis. Since the cameras that are studied in this work (standard cameras and light field cameras) project the light onto a rectangular sensor, it is useful to align the coordinate system's x - and y -axis with the corresponding sensor's s - and t -axis, respectively. Furthermore, due to the almost perspective projection, the change of ray direction with respect to the x - and y -axis should correspond to the change with respect to the s - and t -axis. Thus, using $\mathbf{d}_i = (d_{x,i}, d_{y,i}, d_{z,i})^T$, the rotation angle that aligns both coordinate systems can be found by calculating the mean image gradients with respect to $\mathbf{u} = (s, t)^T$:

$$\begin{pmatrix} d_{xs} \\ d_{xt} \end{pmatrix} = \frac{\sum_i w_i \nabla_{\mathbf{u}} d_{x,i}}{\sum_i w_i}, \quad \begin{pmatrix} d_{ys} \\ d_{yt} \end{pmatrix} = \frac{\sum_i w_i \nabla_{\mathbf{u}} d_{y,i}}{\sum_i w_i}. \quad (5.40)$$

By estimating the orientation angle of the gradients with respect to the sensor axes, a rotation matrix can be found that rotates the coordinate system around the z -axis by an angle α :

$$\alpha_x = \arctan2(d_{xs}, d_{xt}), \quad \alpha_y = \arctan2(d_{ys}, d_{yt}) + \frac{\pi}{2}, \quad (5.41)$$

$$\alpha = \arctan2(\sin \alpha_x + \sin \alpha_y, \cos \alpha_x + \cos \alpha_y), \quad (5.42)$$

$$\mathbf{R}_\alpha = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (5.43)$$

While this gradient-based approach works well for camera systems whose ray surface is a smooth function, problems arise with discontinuities. For light field cameras, the ray direction switches to the opposite direction at the edges of the microlenses. As a result, the gradient shows a strong tendency to the opposite direction, which would lead to a corrupted orientation estimation. However, these too strong gradients can easily be suppressed by means of a threshold value in (5.40), with $w_i = 0$ for $\|\nabla_{\mathbf{u}} d_{x,i}\| > \tau_{\text{thr}}$. And in addition, the weight factor w_i is very small near the microlens edges, due to the higher calibration error ε_i that is caused by the overall worse quality of the optics. And hence, these values are strongly suppressed nonetheless.

After all normalization parameters are found, as the final act, shifting the origin and appropriately rotating the axes transforms the Plücker-ray parameters into the camera-fixed coordinate system. And thus, each ray $\mathbf{l}_i = (\mathbf{d}_i^T, \mathbf{m}_i^T)^T$ is transformed into the new normalized representation:

$$\mathbf{l}_{i,\text{norm}} = \mathbf{T} \mathbf{l}_i, \quad (5.44)$$

with the ray transformation matrix \mathbf{T} that consists of a ray rotation matrix (2.17) and a ray translation matrix (2.18):

$$\mathbf{T} = \mathbf{R}_1 \mathbf{T}_1 = \begin{pmatrix} \mathbf{R}_\alpha \mathbf{R}_z & \mathbf{0} \\ \mathbf{R}_\alpha \mathbf{R}_z [-\mathbf{p}_o]_\times & \mathbf{R}_\alpha \mathbf{R}_z \end{pmatrix}. \quad (5.45)$$

5.3 Calibration of the Reference Target

Besides the camera, also the reference target plays an important role in camera calibration and deflectometry. The commonly used checkerboard

calibration patterns are often printed on paper and then glued to a solid base of wood or cardboard. However, due to this rudimentary construction, the reference target can no longer be assumed to be absolutely flat. This means that the solid base material might be bent and, in addition, small bumps on the paper can locally affect the planarity. The use of monitor screens as reference targets drastically reduces this problem, because the pixel plane has a very high local planarity due to the precise manufacturing process.

As already mentioned, the calibration method presented in this work requires dense features on a reference target, which is why it is recommended to use a monitor as a reference. Nevertheless, monitor screens are not ideal reference targets either. Depending on how they are set up, they can deviate from their ideally planar shape to a greater or lesser extent. Therefore, if this deviation is not sufficiently taken into account, it can lead to a non-ideal calibration. Also, apart from the calibration aspect, if the monitor is placed in a deflectometric measurement setup and, for example, is mounted over the measurement sample, it may show considerable curvature. To prevent this from leading to erroneous measurements, it is therefore imperative that the calibration target is described by appropriate modeling. The modeling of the monitor in this work can be grouped into three sub-aspects, *i.e.*, the modeling of the nonlinear characteristic of the pixel brightness, the modeling of the refraction at the front glass, and the modeling of the screen shape. As briefly mentioned in Sec. 4.1.1, while the brightness characteristic only influences the quality of the registration and can easily be compensated, the two remaining aspects systematically and directly influence the value of the measured coordinates. The coding methods from Ch. 4 explain how a subpixel position in the monitor plane can be assigned to each camera ray employing active illumination. However, only the x - and y -coordinate of the reference point can be determined. So far, it was not specified how the z -coordinate, *i.e.*, the height, can be obtained, or it was implicitly assumed that it is set to zero for a flat monitor.

The following sections deal with the modeling of the reference target, the estimation of the model parameters, and finally the integration of the reference model into the camera calibration framework.

5.3.1 Reference Surface Model

There are several ways to model the non-ideality of the monitor. Bergamasco *et al.* [15] extend their camera calibration algorithm by modeling the influence of refraction at the front glass. They adjust the local monitor coordinates x, y using an additive offset that is calculated using the angle between the ray direction and monitor surface normal. The parameters of the refraction model are predefined and used to improve the camera calibration. Schmalz *et al.* [182] and Chen *et al.* [36] on the other hand, take refraction into account by correcting the z -component of a measured point, whereas the x, y -coordinates remain unchanged. Maestro-Watson *et al.* [127] model the refraction in a similar way, however, they confirm that the monitor surface also deforms the cover glass. Hence, they measure the surface using a coordinate measuring machine to obtain better surface normals for the refraction calculation. Studies by Schmalz *et al.* [182] and Bergamasco *et al.* [15] show that modeling the refraction has only a small impact on applications such as camera calibration or deflectometry. And as investigated by Nüss *et al.* [141], the shape of the monitor has a far greater influence. Bartsch *et al.* [13] model the monitor by representing its surface with a polynomial surface and the model parameters are estimated during the calibration of a deflectometric measurement system. To combine both non-idealities, Reh *et al.* [168] model the z -coordinate of the monitor as an additive superposition of both effects, that is, shape and refraction.

5.3.1.1 Shape Model

Commercially available monitor screens are locally very planar and only deviate globally from the ideal plane, which can be perceived as a slight curvature or torsion. Thus, as suggested by Reh *et al.* [168] and Bartsch *et al.* [13], the z -coordinate of the reference points, *i.e.*, the monitor height, is defined using a bivariate polynomial function

$$z_S(x, y) = \sum_{m=0}^{N_x} \sum_{n=0}^{N_y} c_{mn} x^m y^n, \quad (5.46)$$

where N_x and N_y are the highest orders of the variables x and y , respectively. And the constants c_{mn} represent the coefficients of the correspond-

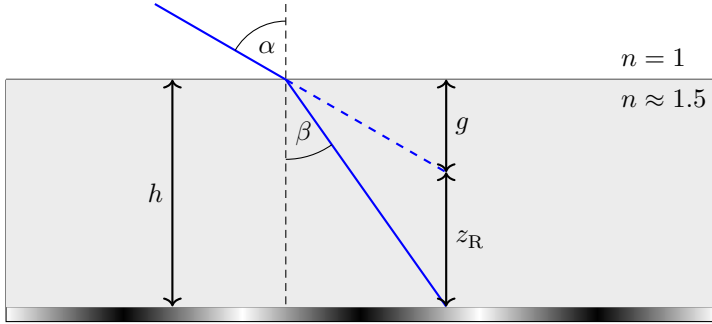


Figure 5.7 Refraction of rays at the cover glass as proposed by Reh *et al.* [168].

ing polynomial components. As shown by Varsamis *et al.* [210], to get a short expression of the bivariate polynomial function and to simplify further calculations, (5.46) is converted into a vector representation:

$$\mathbf{m}(x, y) := [1, x, x^2, \dots, x^{N_x}] \otimes [1, y, y^2, \dots, y^{N_y}] , \quad (5.47)$$

$$\mathbf{c} := [c_{00}, c_{10}, \dots, c_{N_x 0}, c_{01}, c_{11}, \dots, c_{N_x 1}, \dots, c_{N_x N_y}] , \quad (5.48)$$

$$\Rightarrow z_S(x, y) = \mathbf{m}(x, y)^T \mathbf{c} . \quad (5.49)$$

5.3.1.2 Refraction Model

For the modeling of the refraction at the front glass cover, the model of Reh *et al.* [168] and Chen *et al.* [36] shall serve as a reference. The refraction in the cover glass causes the measured monitor coordinates to appear in a slightly closer position, which depends on the angle of incidence of the camera rays. From figure 5.7 follows $h \tan(\beta) = g \tan(\alpha)$ and by using the law of refraction $\sin(\alpha) = n \sin(\beta)$, where n is the refraction index of the glass, it follows

$$\begin{aligned} z_R &= h - g = h - h \frac{\tan(\beta)}{\tan(\alpha)} = h - h \frac{\cos(\alpha)}{n \sqrt{1 - \sin^2(\beta)}} \\ &= h \left(1 - \frac{\cos(\alpha)}{\sqrt{n^2 - 1 + \cos^2(\alpha)}} \right) . \end{aligned} \quad (5.50)$$

To calculate the refraction, the angle α between a camera ray and the surface normal at the observed monitor point must be determined. For the unit normal vector $\hat{\mathbf{n}}$ and a ray with direction vector \mathbf{d} follows

$$\cos(\alpha) = \mathbf{d}^T \mathbf{R} \hat{\mathbf{n}}, \quad (5.51)$$

where \mathbf{R} transforms the monitor coordinate system into the camera coordinate system. While Reh *et al.* [168], for simplicity, consider the refraction model to be completely independent of the shape model, because they use only a very small screen with a diagonal of about 2 cm length, this simplification does not hold in this work. Since commercially available monitors usually have a diagonal of more than 50 cm length, a deformation of the monitor also causes a deformation of the front glass. Therefore, according to Maestro-Watson *et al.* [127], the normal of the front glass should be calculated using the shape model. It follows:

$$\mathbf{n}(x, y, \mathbf{c}) = \begin{pmatrix} -\partial_x z_S \\ -\partial_y z_S \\ 1 \end{pmatrix} = \begin{pmatrix} -\sum_{m=0}^{N_x} \sum_{n=0}^{N_y} c_{mn} m x^{m-1} y^n \\ -\sum_{m=0}^{N_x} \sum_{n=0}^{N_y} c_{mn} n x^m y^{n-1} \\ 1 \end{pmatrix}. \quad (5.52)$$

This leads to the expression for the height deviation caused by the refraction in the front glass

$$z_R(x, y) = h \left(1 - \frac{\mathbf{d}^T \mathbf{R} \hat{\mathbf{n}}(x, y, \mathbf{c})}{\sqrt{n^2 - 1 + (\mathbf{d}^T \mathbf{R} \hat{\mathbf{n}}(x, y, \mathbf{c}))^2}} \right). \quad (5.53)$$

5.3.1.3 Complete Reference Model

As suggested by Reh *et al.* [168], to obtain the reference model, both the refraction model and the shape model are combined. Given the direction of a camera ray \mathbf{d}_i , the point coordinates x_{ik}, y_{ik} that were estimated using phase-shift coding, and the rotation of the reference target \mathbf{R}_k , the value of the z -coordinate can be calculated. Finally, the complete monitor model is represented by $z_C(x_{ik}, y_{ik}) := z_S(x_{ik}, y_{ik}) - z_R(x_{ik}, y_{ik})$, where the z -value of the refraction is subtracted, since the refraction causes the measured monitor coordinates to appear in a slightly closer position.

Using the abbreviations $\mathbf{m}_{ik} := \mathbf{m}(x_{ik}, y_{ik})$, $\hat{\mathbf{n}}_{ik}(\mathbf{c}) := \mathbf{R}_k \hat{\mathbf{n}}(x_{ik}, y_{ik}, \mathbf{c})$, this results in

$$\mathbf{x}_{ik} = \begin{pmatrix} x_{ik} \\ y_{ik} \\ z_C(x_{ik}, y_{ik}) \end{pmatrix} = \begin{pmatrix} x_{ik} \\ y_{ik} \\ \mathbf{m}_{ik}^T \mathbf{c} - h \left(1 - \frac{\mathbf{d}_i^T \hat{\mathbf{n}}_{ik}(\mathbf{c})}{\sqrt{n^2 - 1 + (\mathbf{d}_i^T \hat{\mathbf{n}}_{ik}(\mathbf{c}))^2}} \right) \end{pmatrix}. \quad (5.54)$$

5.3.2 Parameter Estimation

In order to estimate the parameters \mathbf{c} , h and n of the reference model, the newly modeled z -coordinate has to be integrated into the objective function (5.9):

$$f(\mathcal{R}, \mathcal{T}, \mathcal{L}, \mathbf{c}, h, n) = \sum_{i,k} w_{ik} \left\| \begin{pmatrix} \mathbf{R}_k \begin{pmatrix} x_{ik} \\ y_{ik} \\ z_C(x_{ik}, y_{ik}) \end{pmatrix} + \mathbf{t}_k \end{pmatrix} \times \mathbf{d}_i - \mathbf{m}_i \right\|^2. \quad (5.55)$$

Given that the modeling of the front glass results in a strongly nonlinear equation, (5.55) cannot be simplified as demonstrated in the previous sections. Nonetheless, because the monitor model consists of relatively few parameters, it can be optimized using standard gradient descent-based methods (Levenberg-Marquard, BFGS, *etc.* [140]). To ensure the stability of the optimization, the front glass parameters must have constraints to avoid physically unreasonable solutions. The optimal monitor parameters can then be found by solving the following bound-constrained optimization problem:

$$\arg \min_{\mathbf{c}, n, h} f(\mathcal{R}, \mathcal{T}, \mathcal{L}, \mathbf{c}, h, n), \text{ subject to } 1 \leq n, 0 \leq h. \quad (5.56)$$

Since a gradient descent-based optimization is an iterative process, the objective function must be evaluated at least once in each iteration. This leads to the fact that the sum over all rays i and all poses k has to be recalculated very often, which may take several seconds even with an efficient implementation on current GPU hardware. If the monitor optimization is now to be integrated into the generic calibration, the total

optimization time will increase dramatically. However, investigations by Schmalz *et al.* [182] could show that the front glass only has a very small influence on the calibration. It is therefore advisable to estimate only the shape of the monitor and to rely on the manual of the used monitor to obtain the parameters h, n of the cover glass.

If the optimization of the glass cover is omitted, the objective function can be rearranged in a way that the summation over all poses and rays only needs to be evaluated once during the optimization. This results in a very fast optimization. With the help of the abbreviation $\mathbf{a}_{ik} = [\mathbf{d}_i]_{\times}^T (x_{ik}\mathbf{r}_{1k} + y_{ik}\mathbf{r}_{2k} + \mathbf{t}_k) - \mathbf{m}_i$, by using the column vectors of the rotation matrices $\mathbf{R}_k = [\mathbf{r}_{1k}, \mathbf{r}_{2k}, \mathbf{r}_{3k}]$, and by assuming $z_{G,ik} = 0$, the optimization problem (5.55) can be expressed as

$$\begin{aligned}
 f(\mathcal{R}, \mathcal{T}, \mathcal{L}, \mathbf{c}) &= \sum_{i,k} w_{ik} \left\| [\mathbf{d}_i]_{\times}^T (x_{ik}\mathbf{r}_{1k} + y_{ik}\mathbf{r}_{2k} + \mathbf{m}_{ik}^T \mathbf{c} \mathbf{r}_{3k} + \mathbf{t}_k) - \mathbf{m}_i \right\|^2 \\
 &= \sum_{i,k} w_{ik} \left\| [\mathbf{d}_i]_{\times}^T \mathbf{r}_{3k} \mathbf{m}_{ik}^T \mathbf{c} + \mathbf{a}_{ik} \right\|^2 \\
 &= \sum_{i,k} w_{ik} \left\| \mathbf{H}_{ik} \mathbf{c} + \mathbf{a}_{ik} \right\|^2 \\
 &= \mathbf{c}^T \sum_{i,k} w_{ik} \mathbf{H}_{ik}^T \mathbf{H}_{ik} \mathbf{c} + \sum_{i,k} w_{ik} \mathbf{a}_{ik}^T \mathbf{H}_{ik} \mathbf{c} + \sum_{i,k} w_{ik} \mathbf{a}_{ik}^T \mathbf{a}_{ik} \\
 &= \mathbf{c}^T \mathbf{Q} \mathbf{c} + \mathbf{q}^T \mathbf{c} + o. \tag{5.57}
 \end{aligned}$$

An easy-to-find minimum of the above objective function can be obtained assuming that the matrix \mathbf{Q} is positive definite. If so, the optimal parameter vector of the reference model can be straightforwardly inferred without using further optimization steps:

$$\mathbf{c} = 2 \mathbf{Q}^{-1} \mathbf{q}. \tag{5.58}$$

Since the matrix \mathbf{Q} consists of the sum of squares of \mathbf{H}_{ik} , it is positive semidefinite. And due to the objective function being quadratic, a global minimum is obtained. The degenerate case with $\det(\mathbf{Q}) = 0$ occurs in reality only if $\mathbf{H}_{ik} = \mathbf{r}_{3k} [\mathbf{d}_i]_{\times} \mathbf{m}_{ik}^T = \mathbf{0}$ holds for all summands. This means that all camera rays \mathbf{d}_i would have to be orthogonal to the z -axis \mathbf{r}_{3k} of all the reference coordinate systems, *i.e.*, a telecentric camera would

always need to look exactly frontally at the monitor. Because this case can only be achieved for very special imaging configurations, it will not be considered further in this work.

As the estimation of the reference model parameters not only returns the shape of the monitor but also helps to improve the camera calibration, it can be easily integrated into the overall optimization framework. Thus, it is only necessary to calculate the value of the z -coordinate of the reference points using the current reference model (5.54). This is then used in each step of the ray estimation from Sec. 5.2.4 and the pose estimation from Sec. 5.2.5. The alternating minimization for the generic camera calibration can then be extended to a three-step optimization:

$$\mathbf{L}_i^{(n)} = \arg \min_{\mathbf{L}_i \in \mathbb{P}^6} f\left(\mathcal{R}^{(n-1)}, \mathcal{T}^{(n-1)}, \mathbf{L}_i, \mathbf{c}^{(n-1)}\right), \forall i \in \mathcal{I}, \quad (5.59)$$

$$\mathbf{c}^{(n)} = \arg \min_{\mathbf{c} \in \mathbb{R}^{N_x N_y}} f\left(\mathcal{R}^{(n)}, \mathcal{T}^{(n)}, \mathcal{L}^{(n)}, \mathbf{c}\right), \quad (5.60)$$

$$\mathbf{R}_k^{(n)}, \mathbf{t}_k^{(n)} = \arg \min_{(\mathbf{R}_k, \mathbf{t}_k) \in \text{SE}(3)} f\left(\mathbf{R}_k, \mathbf{t}_k, \mathcal{L}^{(n)}, \mathbf{c}^{(n)}\right), \forall k \in \mathcal{K}, \quad (5.61)$$

where the reference target model can be initialized as a flat screen using $\mathbf{c}^{(0)} = \mathbf{0}$. Of course, in order to obtain the complete reference model, the influence of the cover glass on the measured reference coordinates and its parametrization may be included in the overall calibration.

5.4 Calibration of the Deflectometry Setup

While the camera calibration provides a determination of the vision rays and the calibration of the reference target allows modeling of the reference features, for the deflectometric reconstruction of specular surfaces another calibration is necessary: The transformation between camera and monitor coordinates has to be identified to transform the local monitor features into the global camera coordinate system. Here, the assumption is made that the camera and the reference monitor do not move relative to each other so that there is only one transformation. A problem that arises here is that in the deflectometric measurement setup the monitor is generally not in the direct field of view of the camera, since it should only be observed as a reflection on the surface under test. A monitor pose

estimation, as presented in Sec. 5.2.5, does not work here without modification. To estimate the real relative transformation between camera and monitor, one can observe the monitor via the reflection in a reference mirror. If the position and shape of this mirror are known, the virtual (mirrored) monitor pose can be used to determine the true transformation between camera and monitor. In general, however, the position of the mirror is unknown. There are various approaches to solving this difficulty. As probably the most intuitive approach, markers can be placed on the mirror, which allows a direct pose estimation of the mirror [6, 29]. If no markers can be placed on the mirror or if it is not desired that the markers increase the measurement uncertainty, then the mirror pose and the monitor pose can also be calculated indirectly. For this purpose, the mirror is not only placed in one position but in several positions, and the virtual monitor pose is measured each time. The set of virtual poses can then be used to infer the original pose [196, 229, 231].

In this work, the monitor pose is found using a marker-less plane mirror. The following sections explain how the set of virtual poses can be used to obtain a linear solution for the pose. Then, it is described how the generic pose estimation from the previous sections can be used to further improve the linear solution.

5.4.1 Linear Solution

The problem of the deflectometric calibration is shown in figure 5.8. Because the camera does not see the monitor directly but only its reflection, the monitor coordinates \mathbf{x} are first transformed into the camera coordinate system and then reflected at the mirror plane. The virtual coordinates $\tilde{\mathbf{x}}$ can then be calculated with

$$\begin{pmatrix} \tilde{\mathbf{x}} \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{H} & 2d\mathbf{n} \\ \mathbf{0} & 1 \end{pmatrix} \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix} = \begin{pmatrix} \tilde{\mathbf{R}} & \tilde{\mathbf{t}} \\ \mathbf{0} & 1 \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix}, \quad (5.62)$$

where \mathbf{n} is the unit normal of the mirror, d is the shortest distance between the mirror plane and the camera aperture, and $\mathbf{H} = \mathbf{I} - 2\mathbf{n}\mathbf{n}^T$ represents a reflection operator. Depending on the mirror position, the relation

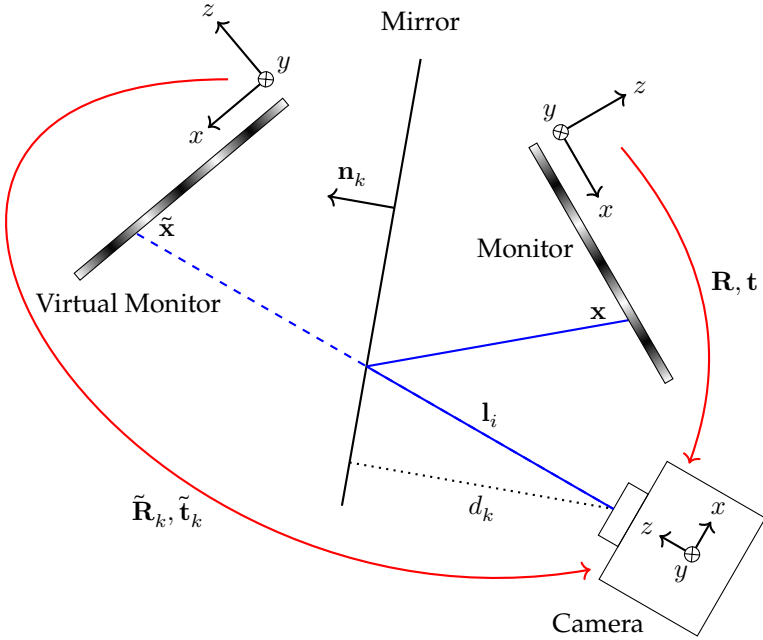


Figure 5.8 Mirror-based pose estimation: The camera sees the reflection of the monitor in the reference mirror. Only the transformation of virtual monitor coordinates to camera coordinates can be estimated.

between the different virtual poses of the reflected monitor and the pose of the true monitor can be directly derived:

$$\tilde{\mathbf{R}}_k = \mathbf{H}_k \mathbf{R}, \quad (5.63)$$

$$\tilde{\mathbf{t}}_k = \mathbf{H}_k \mathbf{t} + 2d_k \mathbf{n}_k. \quad (5.64)$$

To obtain a solvable equation system, the angle of the mirror must be changed for each acquisition. Takahashi *et al.* [196] show that the equations can be solved using an orthogonality constraint if at least three mirror positions are observed. For this, the intersection line \mathbf{m}_{ij} between all possible mirror pairs $i, j \in \{1, 2, 3\}$ is defined. Since the intersecting

lines have to be orthogonal to the respective mirror normals, Xiao *et al.* [229] derive the equation

$$\begin{aligned}\tilde{\mathbf{R}}_i \tilde{\mathbf{R}}_j^T \mathbf{m}_{ij} &= (\mathbf{H}_i \mathbf{R}) (\mathbf{H}_j \mathbf{R})^T \mathbf{m}_{ij} = \mathbf{H}_i \mathbf{R} \mathbf{R}^T \mathbf{H}_j^T \mathbf{m}_{ij} \\ &= (\mathbf{I} - 2\mathbf{n}_i \mathbf{n}_i^T) (\mathbf{I} - 2\mathbf{n}_j \mathbf{n}_j^T) \mathbf{m}_{ij} \\ &= \mathbf{m}_{ij}.\end{aligned}\quad (5.65)$$

The intersection line \mathbf{m}_{ij} can be found as unit eigenvector with the smallest eigenvalue of the matrix $\tilde{\mathbf{R}}_i \tilde{\mathbf{R}}_j^T - \mathbf{I}$. By using the intersection lines, the unit normal vectors of the reference mirror planes can be calculated

$$\mathbf{n}_1 = \frac{\mathbf{m}_{12} \times \mathbf{m}_{13}}{\|\mathbf{m}_{12} \times \mathbf{m}_{13}\|}, \mathbf{n}_2 = \frac{\mathbf{m}_{12} \times \mathbf{m}_{23}}{\|\mathbf{m}_{12} \times \mathbf{m}_{23}\|}, \mathbf{n}_3 = \frac{\mathbf{m}_{13} \times \mathbf{m}_{23}}{\|\mathbf{m}_{13} \times \mathbf{m}_{23}\|}. \quad (5.66)$$

For more than three poses, the normal estimate can also be averaged [197]:

$$\mathbf{M}_i^T \mathbf{n}_i = \mathbf{0} \quad \text{with} \quad \mathbf{M}_i = (\mathbf{m}_{i1}, \mathbf{m}_{i2}, \mathbf{m}_{i3}, \dots), \quad (5.67)$$

where the normal vector is found to be the eigenvector with the smallest eigenvalue of the matrix $\mathbf{M}_i \mathbf{M}_i^T$. Then, using (5.63) and $\mathbf{H}_i \mathbf{H}_i = \mathbf{I}$, a rotation matrix can be calculated for each mirror pose $\mathbf{R}_i = \mathbf{H}_i \tilde{\mathbf{R}}_i$. In the ideal case, all estimates should give the same result. Though, in order to suppress noise, rotation averaging [73] is applied and the mean rotation matrix is calculated using a singular value decomposition:

$$\bar{\mathbf{R}} = \sum_i \mathbf{R}_i \rightarrow \bar{\mathbf{R}} = \mathbf{U} \mathbf{S} \mathbf{V} \rightarrow \mathbf{R} = \mathbf{U} \mathbf{V}. \quad (5.68)$$

Finally, by using (5.64), the remaining translation vector and mirror distances can easily be found by solving a system of linear equations

$$\begin{bmatrix} \mathbf{H}_1 & 2\mathbf{n}_1 & 0 & \dots & 0 \\ \mathbf{H}_2 & 0 & 2\mathbf{n}_2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{H}_N & 0 & 0 & \dots & 2\mathbf{n}_N \end{bmatrix} \begin{bmatrix} \mathbf{t} \\ d_1 \\ d_2 \\ \vdots \\ d_N \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{t}}_1 \\ \hat{\mathbf{t}}_2 \\ \vdots \\ \hat{\mathbf{t}}_N \end{bmatrix} \quad (5.69)$$

Thus, given virtual pose parameters $\tilde{\mathbf{R}}_k, \tilde{\mathbf{t}}_k$, the true pose of the monitor \mathbf{R}, \mathbf{t} can be obtained with a closed-form solution.

To find these virtual pose parameters, in principle, standard PnP methods can be used [109, 133]. However, these usually work with a classical camera model, *e.g.*, the model described in Sec. 5.1. Because this work does not commit to a specific camera model, the generic pose estimation from Sec. 5.2.5 will be used. In the context of a generic camera model, for each camera ray, the distances to the virtual monitor points are minimized:

$$f(\tilde{\mathbf{R}}_k, \tilde{\mathbf{t}}_k) = \sum_i w_{ik} d(\tilde{\mathbf{x}}_{ik}, \mathbf{l}_i) = \sum_i w_{ik} \left\| (\tilde{\mathbf{R}}_k \mathbf{x}_{ik} + \tilde{\mathbf{t}}_k) \times \mathbf{d}_i - \mathbf{m}_i \right\|^2$$

subject to $\tilde{\mathbf{R}}_k \in \text{O}(3)/\text{SO}(3)$

(5.70)

Due to the reflection at the mirror plane, the virtual rotation matrices are now orthogonal matrices with $\det(\tilde{\mathbf{R}}) = -1$. Fortunately, $\text{O}(3)/\text{SO}(3)$ and $\text{SO}(3)$ have the same Lie algebra $\mathfrak{so}(3)$. Therefore, the previously described optimization method can be used and only the initialization has to be adapted. For this purpose, $\tilde{\mathbf{R}}_k \in \text{O}(3)/\text{SO}(3)$ must hold. Since no other information shall be specified, it shall be assumed that the reference mirror is approximately orthogonal to the camera view axis, the LCD surface of the monitor points approximately in the same direction as the camera, and the y -axes of both coordinate systems are approximately collinear. In other words, the monitor coordinate system is rotated by approximately 180° around the y -axis, see figure 5.8. A simple initialization for the virtual rotation matrix is now obtained by defining the virtual pose as a reflection on the y, z -plane $\tilde{\mathbf{R}}_k = \mathbf{I} - 2\mathbf{n}_k \mathbf{n}_k^T$ with $\mathbf{n}_k = (1, 0, 0)^T$. Starting from this, the generic pose estimation can converge sufficiently fast to a solution.

5.4.2 Nonlinear Optimization

The linear solution is usually sensitive to noise, so it is only used as an initialization for a subsequent optimization to refine the monitor pose \mathbf{R} , \mathbf{t} and the positions of the mirror \mathbf{n}_k , d_k simultaneously [229]. To take advantage of the generic camera model, it is advisable to minimize the distance between the observed monitor points and the reflected rays to obtain the optimal transformation between the camera and monitor coordinate system and, in addition, to obtain the mirror pose. Hence, to

simplify the optimization, the mirror pose parameters are combined into one vector [227]:

$$\mathbf{v}_k := d_j \mathbf{n}_k \Rightarrow \mathbf{H}_k = \mathbf{I} - 2 \frac{\mathbf{v}_k \mathbf{v}_k^T}{\|\mathbf{v}_k\|^2}. \quad (5.71)$$

Since in the deflectometric test setup the monitor is often suspended above the test sample, it shows a non-negligible curvature due to gravity. Therefore, it makes sense to estimate the monitor parameters \mathbf{c} at the same time. The minimization of the following distance measure provides the desired parameters

$$f(\mathbf{R}, \mathbf{t}, \mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{c}) = \sum_{i,k} w_{ik} \|(\mathbf{H}_k(\mathbf{R}\mathbf{x}_{ik}(\mathbf{c}) + \mathbf{t}) - 2\mathbf{v}_k) \times \mathbf{d}_i - \mathbf{m}_i\|^2. \quad (5.72)$$

Since this objective function is highly nonlinear, but also contains only a few parameters, it can be minimized using standard optimization methods, *e.g.*, BFGS or Levenberg-Marquardt [140].

The calibration result could be further improved by performing a holistic calibration and by including the calibration of the camera rays in the optimization [6]. However, there is the problem that the reference mirror must be very accurate and very planar. Because of this, sufficiently large mirrors are not available or are extremely expensive (*e.g.*, a highly planar mirror with only 10 cm² diameter already costs about 800 €). If a standard model-based camera calibration is used, this is not a big concern, because not every ray necessarily has to observe a reference feature. For generic methods, however, it must be ensured that each ray can observe enough points in the monitor plane. When only a small mirror is available, the calibration procedure is time-consuming as the mirror has to be placed in several positions. Therefore, a holistic optimization will not be considered further in this work and is only given as a brief outlook.

5.5 Evaluation

The following sections examine the steps necessary for system calibration and analyze the presented procedures. For the evaluation, a 27" monitor with a resolution of 2560 × 1440 px and a pixel pitch of 233 μm

was used to display the necessary calibration patterns. Two different imaging systems were used to evaluate the proposed generic camera calibration: A standard webcam (Logitech C920 HD Pro Webcam) and a microlens array-based light field camera (Lytro Illum). The first represents a central camera that can be modeled with the classical pin-hole camera approach, whereas the second camera ultimately results in a non-central camera with multiple projection centers, which in addition requires a much more complex camera model to be efficiently calibrated. The monitor was captured from 20 different poses, whereby several phase-shift patterns have to be recorded at each pose to encode the target features. The phase-shifting was performed the same in both horizontal and vertical direction with $M = 12$ shifts per sequence and with the frequencies $\mathbf{f} = (1, 4, 16, 64)$, corresponding to wavelengths of $\lambda = (2560, 640, 160, 40)$ pixels. The distances between monitor and camera were in the range of 5 cm to 2 m. To compare the proposed technique to the classic methods, the webcam was calibrated using the pinhole model of Sec. 5.1 and Zhang’s algorithm [246], which is implemented in the OpenCV library [26]. The light field camera was calibrated using the state-of-the-art method by Bok *et al.* [20]. Both methods use static checker patterns that were displayed on the reference monitor. In addition, the calibration is also performed with the state-of-the-art generic calibration method from Bergamasco *et al.* [15]. They calibrate the camera by iteratively calculating the intersection of the rays with the monitor plane, and by minimizing the resulting coding error to the observed target features on a pixel level. In addition, they estimate the reference pose using an adapted iterative closest point method, where they calculate the perpendicular projection of the 3D reference features onto the corresponding rays, and then align the set of 3D features with the set of perpendicular projections in an iterative manner.

Because the webcam has a smooth mapping from pixels to camera rays, the spatio-temporal phase unwrapping from Ch. 4 can be used without any restrictions, which allows mapping reference features to camera pixels. However, when the light field camera is used, strong discontinuities appear near the edges of the microlenses. Consequently, these edges need to be detected using the edge detection presented in Sec. 4.4.4. And as a result, for these edge pixels, only the temporal unwrapping should

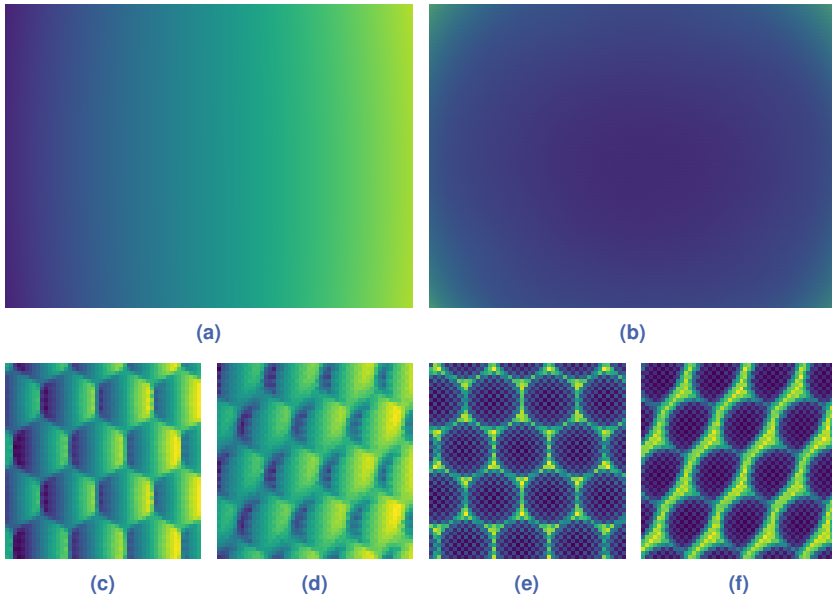


Figure 5.9 Reference feature acquisition for the Lytro Illum camera: (a) shows the encoding of the monitor's x -coordinate. (b) shows the coordinate uncertainty. (c) & (d) show details of the x -coordinate. (e) & (f) show detailed views of the coordinate uncertainty. (c) & (e) show the center region and (d) & (f) the bottom right region of the camera sensor. For better visualization, the color maps are stretched to maximize the contrast.

be used. Figure 5.9 shows the acquisition of reference features for the Lytro Illum camera using phase-shift coding with probabilistic phase unwrapping. It can be seen that the phase measurement shows strong discontinuities near the boundaries of the microlenses. Also, in these areas, the uncertainty increases due to vignetting that is caused by the main lens and by the microlenses. This effect increases even more the closer the pixels are to the edge of the sensor. In addition, the Bayer pattern of the camera sensor affects the uncertainty in such a way that it increases for the red and blue pixels (because in this specific dataset, the spectrum of the displayed pattern seems to be centered around the central wavelength of the green pixel).

5.5.1 Error Metrics

To ensure a fair comparison between the calibration methods, the different models are examined with regard to their point-to-ray distance of each ray to every observed feature in every monitor plane

$$\varepsilon_{ik} := \|\mathbf{p}_{ik} \times \mathbf{d}_i - \mathbf{m}_i\| . \quad (5.73)$$

For the error metrics, the mean distance and the root-mean-squared error (RMSE) are calculated:

$$\text{Mean}(\varepsilon) := \frac{1}{\sum_{ik} w_{ik}} \sum_{ik} w_{ik} \varepsilon_{ik} , \quad (5.74)$$

$$\text{RMSE}(\varepsilon) := \sqrt{\frac{1}{\sum_{ik} w_{ik}} \sum_{ik} w_{ik} \varepsilon_{ik}^2} . \quad (5.75)$$

The comparison is here done using a weighted distance (with $w_{ik} = \sigma_{ik}^{-2}$) that allows to assess the quality of the camera calibration without being too dependent on the quality of the used reference target features. For a demonstration of the benefit of using additional uncertainty information, the Euclidean distances are evaluated too by defining $w_{ik} = 1$. In the following, weighted distances are symbolized by the variable ε_w , and Euclidean distances by the variable ε_e . A comparison of the commonly used projection error on a pixel level is not possible, because in a generic camera model there is nothing like an “image plane” – there is just a set of rays.

5.5.2 Initialization of the Alternating Minimization

In principle, the presented generic calibration procedure can be initialized using the model-based approaches. For the webcam, the standard camera calibration and pose estimation provided by the OpenCV framework can be used. And for the light field camera, the calibration by Bok *et al.* with a succeeding standard pose estimation may help in the initialization.

As a more generic alternative, one could also initialize using the generic relative pose estimation algorithm proposed by Ramalingam *et al.* [165]. The disadvantage of the method is, however, that the underlying camera model must be known. Different algorithms are needed for the central

model of the webcam and the non-central model of the light field camera. In addition, the use of a two-dimensional planar calibration target (instead of a 3D target) adds ambiguities, which can be resolved only if there is a rough knowledge of the poses. Simulations and experiments showed that their method works in principle, but that the procedure is highly susceptible to noise. However, there are very severe complications for only slightly non-central cameras, like the MLA-based light field cameras used in this work. As described by Ramalingam *et al.* [165] too, the procedure becomes extremely unstable, and no reliable pose can be estimated, even if only a very small noise is present. For light field cameras, the method is therefore rather unusable and will thus not be considered further in this work.

Nevertheless, because using another calibration procedure increases the overall effort, it would be best to rely only on the here presented generic calibration method. In this context, it could be observed that in many cases it was even acceptable to just "guess" the initial positions of the monitor. For example, although the monitor poses in figure 5.10(a) are randomly initialized, the optimization converges towards the optimal solution. However, even if the alternating minimization is strictly convergent, when using a random initialization, with some starting configurations it becomes possible that the optimization gets stuck in suboptimal solutions. Figure 5.10(d) depicts this situation, where some monitor poses are estimated to lie behind the camera. To further minimize the error, the algorithm causes all monitor poses to lie flat on top of each other, and to eventually have the same rotation. The estimated ray bundle is then slit-shaped and completely flat, which is not the correct solution. To avoid such problems, investigations showed that it helps to properly initialize the translation vector of the monitor poses in such a way that the order of distances between camera and monitor poses is approximately correct. Hence, it is useful to specify the distance to the camera during the data acquisition for a subset of monitor poses, so that the distance is approximately known. *E.g.*, the first three monitor poses could be placed about 10 cm apart. Using only this subset of monitor poses, the ray parameters can be estimated with sufficient accuracy in only 20-30 iterations. And finally, this rough estimation of rays can be used to initialize the camera

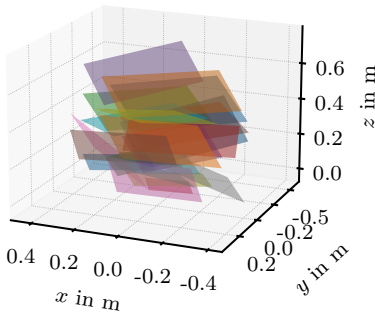
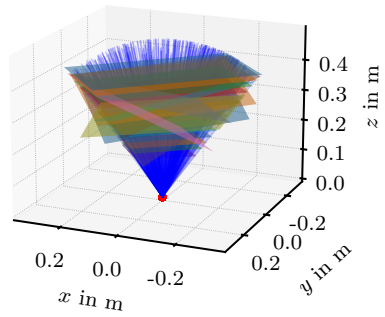
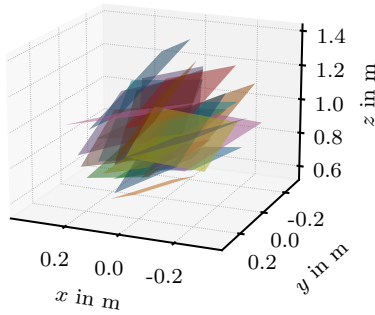
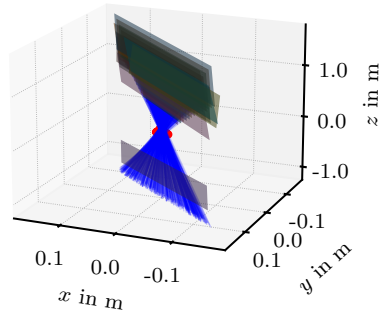
(a) Iteration 0: RMSE ≈ 15 cm.(b) Iteration 1000: RMSE ≈ 32 μ m.(c) Iteration 0: RMSE ≈ 14 cm.(d) Iteration 1000: RMSE ≈ 2250 μ m.

Figure 5.10 Initialization and final result: The figures show the observed monitor area and the calibrated camera rays at start and end. Note the difference in scale. (a) & (b) Even with an initially very bad pose estimation, the procedure converges towards reasonable results. (c) & (d) A badly chosen initialization may converge to a suboptimal local minimum.

calibration for the complete set of poses, where the remaining poses can of course be positioned arbitrarily.

5.5.3 Convergence of the Alternating Minimization

Figure 5.11 shows the convergence of the proposed method as a function of the weighted RMSE of the calibration error over the number of iterations. Here, the calibration was carried out with and without acceleration and with and without modeling the reference monitor. To investigate

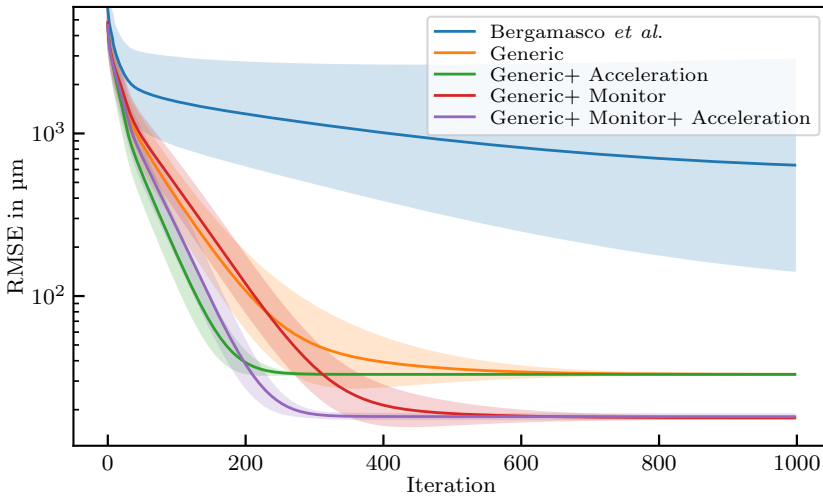


Figure 5.11 Convergence of AM-calibration depending on the initialization: The plot shows the mean value and the $\pm\sigma$ -range of the convergence of the objective function for various initializations.

the robustness against a bad initialization, the convergence behavior was investigated for 50 trials while random translations in the range ± 10 cm per direction and random rotations $\pm 10^\circ$ per axis were added to the starting pose. For comparison, the convergence behavior of the generic calibration method of Bergamasco *et al.* is also investigated, using the same initializations. Although they minimize a different metric in their optimization, the point-to-ray distance is evaluated here after each iteration so that a fair comparison can be made.

The plot shows the average and the standard deviation of the RMSE over all trials, visualized by the thick line and the light background color. Figure 5.11 shows that the proposed method converges significantly faster than the method of Bergamasco *et al.*, and it shows that it is less sensitive to a bad initialization, which is shown by the smaller standard deviation in the error. Starting from some initialization, the method of Bergamasco *et al.* leads to suboptimal solutions. The presented methods show slightly different behaviors in the convergence during the minimization, yet every trial converges very close to the same solution, visible by the very

low standard deviation in the last iterations. In addition, the improved convergence rate when using the Nesterov acceleration is clearly visible. Hence, the minimization converges to a sufficiently accurate result after about 300 iterations. And finally, it can be well acknowledged that the monitor model can push the total calibration error even further down. Interestingly, when estimating the monitor model, it can be observed that the convergence rate is slightly worse than compared to when it is not estimated. This can be explained by considering that the alternating minimization now consists of three subproblems, and with the increasing number of subproblems the convergence rate decreases.

Since each ray is independent of one another, it is possible to process them in parallel, using a GPU. The optimization of 40 million pixels (Lytro Illum) and 20 reference poses then only takes a few seconds per iteration (Intel Core i7-6700, Nvidia GTX 1080 Ti, 16GB RAM). Therefore, the overall calibration for the light field camera converges after about 45 minutes. The presented generic method is even faster and converges after only a few minutes when calibrating the two-megapixel webcam.

5.5.4 Required Number of Poses

Theoretically, a ray can be estimated with two different point observations (then \mathbf{A}_{dd} is positive definite). And to fit a pose, three non-parallel rays are needed (then \mathbf{A}_{tt} is positive definite) that observe different points (then \mathbf{A}_{rr} is positive definite). With only two reference targets, the optimization always converges to a perfect fit, which of course is useless. An unambiguous and correct solution, however, can theoretically be obtained with at least three reference poses [165]. But of course, because the presented calibration is based on a least-squares minimization approach, and because the impact of noise should be reduced, more reference targets are necessary. This becomes apparent in figure 5.12 that shows the calibration error when different numbers of reference targets are used. For this purpose, the camera was calibrated 100 times, where each time a fixed number of target patterns was randomly selected from a total set of 60 poses. The mean error of all calibrations and their $\pm\sigma$ standard deviation are plotted over the number of used patterns. It can be seen that the overall calibration error needs at least a minimum of 15–20 poses to result in a good calibration, whereas more poses increase the overall robustness of the

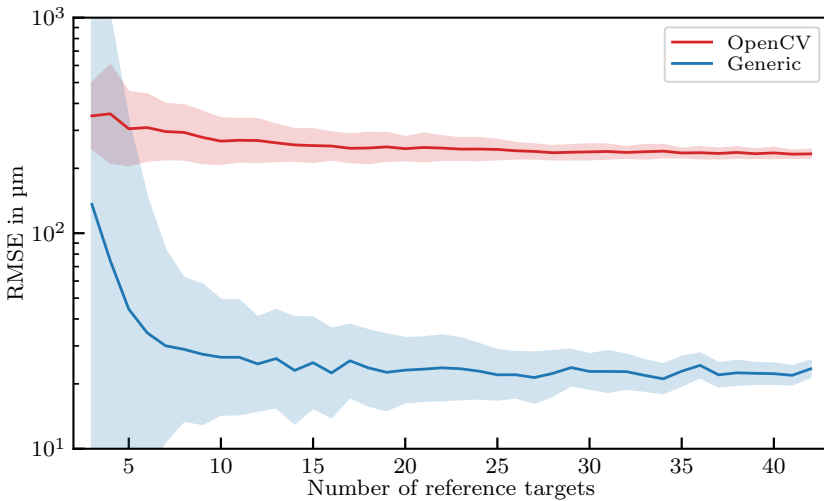


Figure 5.12 Dependency on the number of patterns: The plot shows the mean value and the $\pm\sigma$ -range of the error.

method. Too few patterns, on the other hand, result in a very unreliable calibration. One can see similar results for the OpenCV calibration, although the dependency on the number of patterns is not as strong as compared to the proposed method. In summary, the proposed calibration needs more reference poses to correctly estimate the immense number of parameters. However, even with fewer poses, the error of the proposed calibration is several times smaller than the model-based calibration.

5.5.5 Evaluation of the Calibration Error

For a quantitative comparison of the different calibration methods, the calibration error will be compared in the following. To verify the positive influence of using the reference target uncertainty on the calibration, the method “Generic (E)” is investigated in addition. This method is the same as the presented method but does not use the uncertainty, and instead only minimizes the Euclidean point-to-ray distance by defining $w_{ik} = 1$ for all target features. For a faster calibration, the proposed methods use Nesterov acceleration. In addition, the webcam was calibrated with

Table 5.1 Calibration of the Logitech webcam.

Logitech Webcam	ε_w in μm		ε_e in μm	
	Mean	RMSE	Mean	RMSE
OpenCV (checker)	–	–	304.9	395.0
OpenCV	366.1	415.9	369.7	424.6
Bergamasco <i>et al.</i>	91.4	109.6	92.0	110.0
Generic (E)	25.9	33.7	28.2	37.6
Generic (H)	23.7	33.0	27.9	41.9
Generic	23.6	32.4	27.8	39.4
Generic + Monitor model	14.7	17.6	14.9	17.7

OpenCV and the light field camera with the method of Bok *et al.*, where checkerboard features were used as reference. For methods with the suffix “checker”, the error was evaluated only for those camera pixels that see the detected checker features. The other methods were evaluated for each camera pixel using the phase-shift features.

5.5.5.1 Webcam Calibration

Table 5.1 summarizes the result of the webcam calibration for the different algorithms. It can be seen that the presented generic methods produce the best results. Even for the webcam, with its relatively simple optics, the presented method delivers both a smaller mean error and a smaller RMSE for both error metrics, resulting in a more precise geometric calibration with fewer outliers at the same time. In the classic model from the OpenCV library, most outliers cannot be used because they are too far away from the model description. The generic model can effectively use each individual pixel as a source of information. This becomes particularly visible for the OpenCV calibration when only the error regarding the checkerboard features is evaluated. Here, the error is smaller than when for every pixel all phase-shift features are evaluated. This demonstrates that the classic calibrations optimize the camera model for only a part of the pixels, namely the ones that observe checker features. The remaining pixels are interpolated through the camera model and thus have a larger calibration error. Figure 5.13 shows the error per pixel

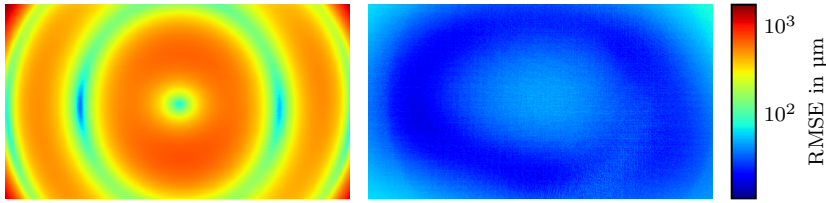


Figure 5.13 Calibration error of the webcam: The OpenCV calibration on the left shows strong systematic errors due to the parametric modeling approach, while the generic model on the right shows a more noise-like result.

for the standard calibration and the presented generic one. It can be seen that the OpenCV calibration shows significant systematic errors, as the error increases or decreases depending on the distance to the center of the sensor. This wave-like behavior of the error is caused by the insufficient modeling capability of parametric models. Even for a simple webcam, the parametric modeling approach does not lead to perfect results. On the other hand, the generic approach calibrates each pixel individually, and hence, almost no systematic errors appear. The resulting calibration error is overall much smaller and has an almost noise-like characteristic.

The proposed methods also perform better than the generic approach by Bergamasco *et al.* Even if the uncertainties are not taken into account and only the Euclidean distance is minimized, the presented method still outperforms the method by Bergamasco *et al.* Moreover, it can be seen that additional information about the coordinate uncertainty further improves the calibration. Inaccurate points are weighted less strongly and therefore have a weaker effect on the result. Interestingly, because “Generic (E)” directly minimizes the Euclidean RMSE, the respective value is smaller than the same metric for the “Generic” method. However, the corresponding mean value of the uncertainty-based method is smaller, since outliers have less influence on the calibration. When using a hierarchical phase unwrapping approach with “Generic (H)”, the mean error slightly increases, although the used phase-shift coding with $M = 12$ shifts already strongly reduces the noise. The corresponding RMSE values increase slightly more than compared when the probabilistic unwrapping is used in “Generic”, meaning that outliers are caused by errors in the hierarchical phase unwrapping. Finally, using the monitor

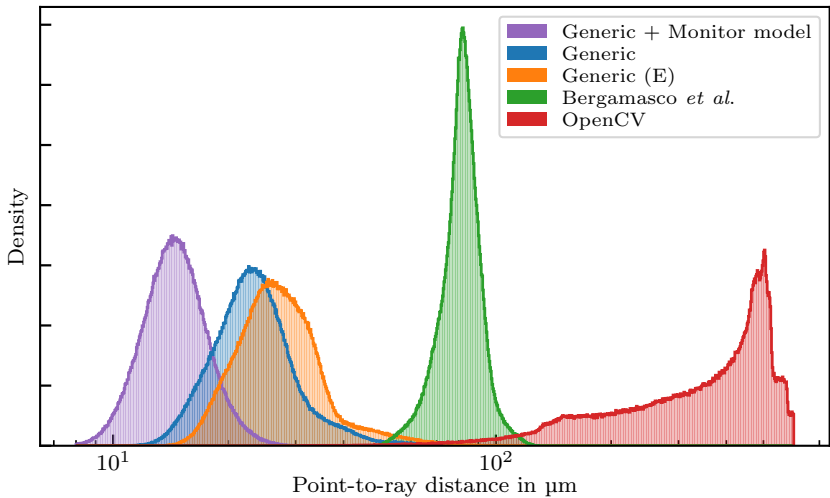


Figure 5.14 Histogram of point-to-ray distances concerning all phase-shift features (Logitech Webcam). The generic model creates a much tighter distribution with fewer outliers as compared to the classical calibrations (note the logarithmic scale). Outliers can be further suppressed with uncertainty information and by modeling the reference monitor.

model and estimating its parameters reduces the overall calibration error even more.

Figure 5.14 illustrates the results of the calibrations by showing the distribution of all point-to-ray distances. The OpenCV calibration shows a very widespread distribution that is not symmetric due to the systematic modeling errors. The error distributions of the generic approaches are tighter, shifted to lower values, and are close to a normal distribution, which is to be expected since the errors are calculated from the set of independently calibrated rays.

5.5.5.2 Light Field Camera Calibration

Similar conclusions can be drawn with the Lytro Illum light field camera. Table 5.2 summarizes the results of the calibration for the different algorithms. Due to the more complex optics and the more extensive optimization associated with this camera, the differences here are much

Table 5.2 Calibration of the Lytro Illum camera.

Lytro Illum	ε_w in μm		ε_e in μm	
	Mean	RMSE	Mean	RMSE
Bok <i>et al.</i> (checker)	–	–	448.4	851.8
Bok <i>et al.</i>	375.6	758.8	7165.1	8686.0
Bergamasco <i>et al.</i>	922.1	1696.5	1041.8	1720.5
Generic (E)	77.1	185.3	163.1	438.5
Generic (H)	60.4	153.4	165.2	510.7
Generic	49.4	105.8	155.5	457.4
Generic + Monitor model	43.3	91.4	149.8	439.7

greater and the superiority of the proposed generic calibration becomes even clearer. Although the model by Bok *et al.* is very sophisticated, it is adapted strongly to the few checkerboard features and only produces good results here. But if the same model is evaluated for all phase-shift features for every pixel, then this leads to huge RMSE values caused by many outliers. In this case, one can see particularly well that a low dimensional model-based approach cannot ideally describe every pixel of a camera with complex optics, such as the light field camera. Moreover, the benefit of using uncertainties becomes very well apparent: the quality of pixels in microlens-based light field cameras (and the ability to model the corresponding rays accurately) deteriorates towards the edges of the microlenses, leading to increased uncertainties (see figure 5.9). These can however be suppressed effectively by the proposed generic method, leading to much smaller mean errors and RMSE values for both error metrics.

The method by Bok *et al.* can calibrate the center of each microlens very well. Here, their calibration error reduces to about $60\mu\text{m}$ for the best pixels. This results in a relatively good reconstruction of the central sub-aperture image, as will be analyzed in detail in Ch. 6. However, the more the pixels move away from the microlens center, the larger becomes the error. This reduces the overall calibration quality, as shown in the results. Also, the method by Bok *et al.* returns a light field with only 35 million pixels, as compared to the total of 41 million pixels of the raw data. The worst pixels, which are between neighboring microlenses, are not used

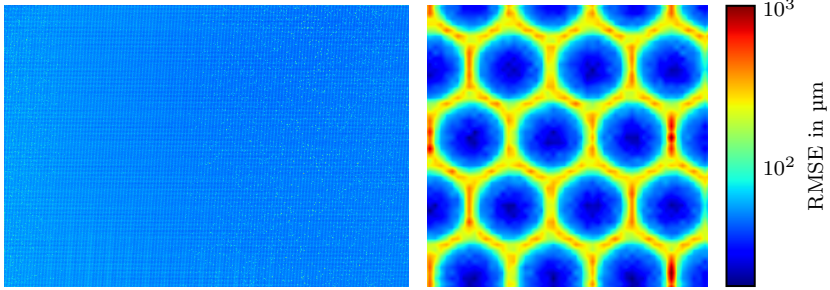


Figure 5.15 Calibration error of the Lytro Illum. Left: At a global view, the error is independent of the position on the sensor. Right: The error increases near the microlens edges.

in the modeling and are therefore cut off. Thus, they cannot be analyzed in the evaluation made here. However, the proposed generic model can effectively calibrate the rays of every pixel of the sensor, whereby not only good calibration results in the centers of the microlenses are achieved, but also at the edges, where it is very difficult to describe the light field camera with a uniform model. By using the uncertainty of the target features, these pixels at the microlens edges can be easily identified as outliers. Therefore, they are automatically compensated and have less influence on the pose estimation, which further improves the ray estimation. Interestingly, using the hierarchical phase unwrapping to obtain the monitor coordinates with “Generic (H)” instead of the probabilistic approach with “Generic” has a more significant effect on the light field camera than it had on the webcam. The overall calibration error is much larger, which is caused by the pixels at the microlens edges. Here, due to the strong vignetting, the signal-to-noise ratio is reduced, resulting in higher phase noise. This again further demonstrates the advantages of the proposed probabilistic phase unwrapping. Finally, using the monitor model and estimating its parameters further reduces the overall calibration error. When compared to the webcam calibration, the improvement here is smaller.

Figure 5.15 shows the calibration error of the proposed generic method for each camera pixel. Although the error increases near the microlens edges, it is still very small. The reason that these pixels cannot be described better by the generic camera model is that in reality there is a

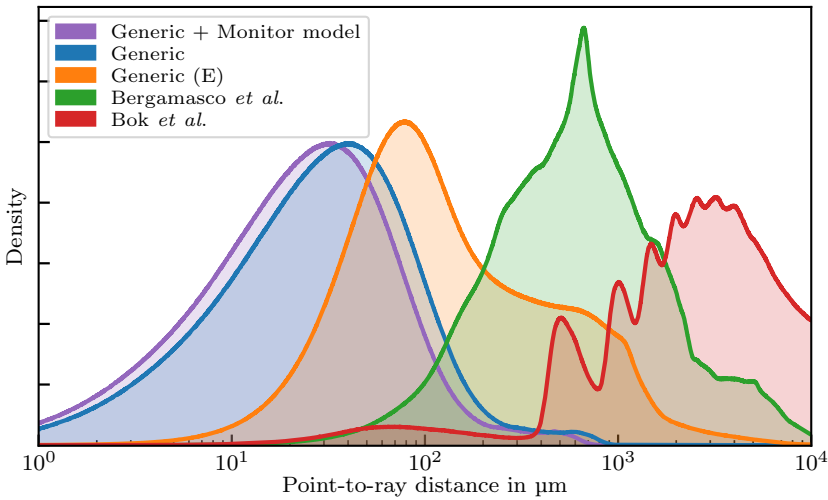


Figure 5.16 Histogram of point-to-ray distances concerning all phase-shift features (Lytro Illum). The generic model creates a much tighter distribution with less outliers as compared to the classical calibration (note the logarithmic scale). Outliers can be suppressed even more with uncertainty information and by modelling the reference monitor.

superposition of vision rays. This means that the light cone belonging to the ray is either strongly elliptically distorted or simply consists of the superposition of multiple individual cones. A disadvantage of the generic camera model is that it can only return the mean value of the corresponding light cone for such pixels, which does not necessarily reflect reality. Nonetheless, because a superposition of light cones causes a high uncertainty in the phase-shift coding, the corresponding pixels have an insignificant influence on the uncertainty-based calibration presented here. Especially during the pose estimation, outliers are strongly suppressed, and thus, the overall calibration result still turns out well.

While the method by Bergamasco *et al.* delivers good results for the webcam, it does not seem to work well with the light field camera. Although the calibration of the webcam shows that their approach works, it seems that it does not generalize as well as the proposed method and that it has difficulties with the poor quality of the pixels at the edges of the microlenses. The procedure diverged in the experiments. Only after im-

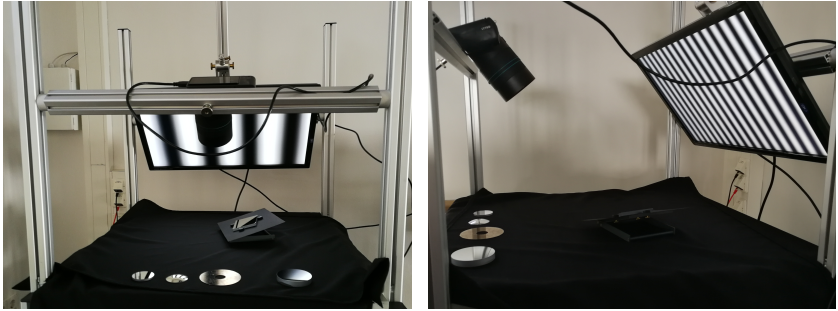


Figure 5.17 Experimental deflectometry setup.

proving the initialization for a few iterations using the proposed method and by excluding the pixels with the highest uncertainty, a convergent result for their method could be obtained, which still has a smaller error than the calibration by Bok *et al.*

Figure 5.16 summarizes the results and shows the distribution of all point-to-ray distances for the different calibrations of the Lytro Illum. The method by Bok *et al.* results in a multi-modal distribution with the lowest peak at about $60\text{ }\mu\text{m}$. Also, several peaks systematically appear at higher distances, which is due to the difficulties of modeling a light field camera. These peaks correspond to the average calibration error of the individual sub-aperture images, as will be discussed in detail in Ch. 6. The method by Bergamasco *et al.* results in a distribution with many errors at high values (more than 1 mm). The proposed methods, on the other hand, are much tighter with peaks at far lower values. Moreover, larger errors from minimizing only the Euclidean distance can be reduced to smaller ones by using the generic calibration with uncertainty-based weighting. And in addition, using a monitor model further improves the result.

5.5.6 Mirror-Based Pose Estimation

The experimental setup of the deflectometry system used in this work is shown in figure 5.17. The reference monitor is the same as the one used for the camera calibration, and the camera is the Lytro Illum, which was calibrated using the generic calibration. To perform a deflectometric

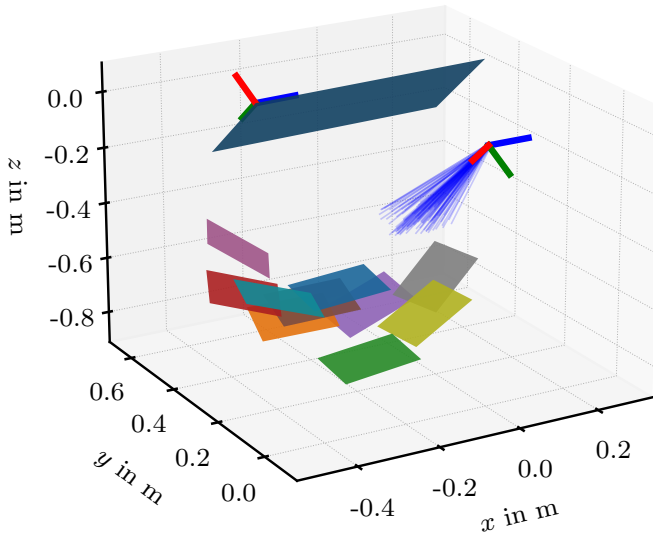


Figure 5.18 Result of mirrored pose estimation for the experimental setup from figure 5.17. Top left: Reference monitor. Top right: Camera and estimated camera rays. Bottom: Reference mirrors.

measurement, the relative pose between camera and monitor must be calculated using the procedure from Sec. 5.4. For this purpose, a precision surface mirror with $\lambda/20$ flatness is used as a reference mirror, *i.e.*, with the reference wavelength of 632.8 nm, the mirror has a maximum peak-to-valley deviation from the perfect plane of 31.64 nm. The mirror is placed in 10 different positions and each time reference points are recorded using phase-shift coding. Using the adapted generic pose estimation (5.70), the mirrored pose of the virtual monitors is estimated. Then, using the procedure shown in Sec. 5.4.1, a linear solution is found for the true pose between the camera and monitor. Subsequently, the non-linear optimization (5.72) improves the final estimate. Figure 5.18 shows the mirror planes and the resulting pose of camera and monitor. Interestingly, the calibration error varies considerably during the estimation steps. The estimation of the virtual poses results in an RMSE of 92.0 μm . However, after the linear solution for the true pose is found, it increases

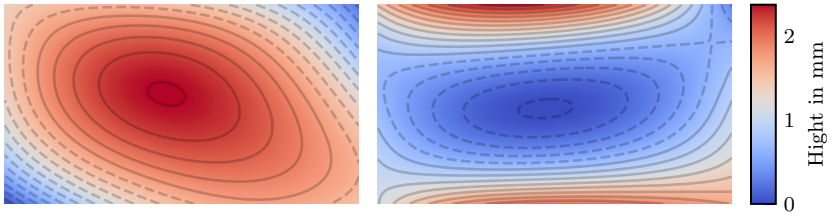


Figure 5.19 Visualization of the display surface. Left: Because of the weight on the corners, the monitor surface is twisted. Right: The monitor hangs above the surface and due to gravity, the surface is slightly bent.

substantially to $640.2\text{ }\mu\text{m}$. An explanation for this is that when two mirror positions are only slightly inclined to each other, the distance of the intersection line of both planes to the measurement setup can grow to very large values, which leads to numerical instabilities in (5.67) and (5.69). Still, the subsequent nonlinear optimization can compensate for this, so that the RMSE of the final pose estimation decreases again to $95.2\text{ }\mu\text{m}$.

5.5.7 Shape Estimation of the Reference Target

It could already be shown that the monitor model improves the calibration. However, it is not yet clear whether the model also provides realistic values. To verify this, the monitor was measured in two positions. In the first measurement, the monitor lies on the ground, and both the upper left corner and the lower right corner are loaded with weights. Hence, the monitor should show a torsion. A second measurement shows the monitor in the deflectometry setup, see figure 5.17. Here, the monitor hangs above the surface under test, and the screen points downwards. This causes the outer areas of the monitor to also bend downwards, which results in an increased curvature of the display surface. The monitor parameters can be obtained after calibration, and with them, the shape of the monitor can be calculated. Figure 5.19 shows the results for both measurements. The figure shows very well that the first monitor has a strong torsion, while the second one is slightly curved, as was to be expected. The distance between the highest and the lowest point for the first measurement is 2.5 mm and for the second measurement 0.8 mm .

This was also approximately verified by placing a straight metal bar on the surface and by measuring the distance between the bar and the screen surface with a ruler. Therefore, in conclusion, the calibration of the monitor shape is satisfactory.

5.6 Summary

In this chapter, the calibration of the deflectometric measuring system was described. The main contribution was a new calibration technique for the generalized camera model. The proposed method splits the calibration into two parts, a ray calibration and a pose estimation, and it applies an alternating minimization to efficiently optimize the immense number of parameters. Dense calibration features were obtained using phase-shift coding techniques, and the measurement uncertainty that was estimated during the pre-processing could be used in the optimization. A simple analytical solution to minimize the ray subproblem was presented. Further, the pose was optimized by decoupling rotation and translation, and by using gradient descent on the rotation manifold. Since calibration references, *i.e.*, standard LCD screens, are generally not ideal, the shape and also the refraction at the cover glass were modeled, which allowed the estimation of the reference parameters to be efficiently integrated into the generic calibration. Because alternating minimization typically has a slow convergence rate, Nesterov's acceleration scheme was modified to speed up the optimization process. Since in a deflectometric measurement setup, the reference monitor is not in the camera's direct field of view, a mirror-based pose estimation was adapted, which further could be efficiently combined with the presented generic calibration procedure.

Finally, experimental evaluation verified the advantages of the proposed camera calibration method over conventional and other generalized approaches. In this context, the benefit of using additional information about the uncertainty of the calibration target coordinates was demonstrated, and it could be shown that modeling the reference target leads to a considerable improvement in the calibration.

6 Light Field Reconstruction

The calibration methods from the last chapter can already describe all of the optical components very precisely. The generic camera model achieves a high degree of accuracy, but in the process of the calibration, information is discarded, namely, the topological relations between the pixels. For many areas of optical metrology, this does not pose a problem, as often only the geometric ray properties are relevant [145, 209, 247]. In profilometry, for example, a projector illuminates a scene with a coded pattern sequence and each scene point can thus be assigned to a projection ray and a vision ray, allowing for a direct triangulation of the point's depth. The same principle cannot be implemented in deflectometry without further work, since it is not the specular object that is optically encoded here, but the distorted mirror image of the reference pattern generator. Therefore, direct triangulation of the surface cannot be performed. Rather, the object is measured indirectly by triangulation of the normal field, as will be explained in Ch. 7. An important step in this triangulation is the forward and backward projection from camera rays to 3D points and vice versa. While it is very easy to calculate the 3D points along the corresponding ray for each pixel, it is very difficult with the generic camera model to find out to which pixel a 3D point is projected. More specifically, it would be extremely time-consuming to calculate for each 3D point its closest camera ray (or rays), since a complete search over all rays would have to be performed for each point. For an indirect triangulation of the specular surface, the completely generic camera model is therefore unsuitable. Hence, further processing of the calibrated rays is required to recover the neighborhood information between the pixels or, in the case of the light field camera, it is necessary to restore the 4D relation between the camera rays.

Apart from the difficulties that arise in deflectometry, light field cameras also have many other applications, where the geometric calibration of the camera itself is not crucial, but rather the correct reconstruction of

the light field and its SAIs. Examples of this are depth estimation, changing the perspective on the scene, digital refocusing and artificial bokeh, or hyperspectral image reconstruction, as discussed in Sec. 3.2. The emphasis is therefore rather on the reconstruction of the image content than on the ray parameters. To use light field cameras for such applications, it is necessary to obtain the 4D information of the light field. And this information must first be decoded from the raw 2D sensor data of the respective light field cameras. Unfortunately, due to their complex structure, their calibration is very difficult and usually precisely tailored to the particular type of light field camera. Hence, specially adapted algorithms have to be used and a great deal of effort must be invested in modeling the camera optics. However, as already described in the last chapter, low-dimensional models are often not sufficient to represent all properties of an optical system—especially when it comes to sophisticated and highly specialized optical systems like light field cameras. In fact, the characteristics of the optics are already very precisely incorporated in the generic ray bundle. Therefore, it makes sense to directly utilize the calibrated rays for light field decoding as well.

To overcome the issues of highly specialized decoding algorithms, and to use the already precisely estimated camera rays, this chapter presents an algorithm that uses the generic camera calibration as a basis for reconstructing a light field from the unconstrained set of rays. Hereby, a generic light field reconstruction is realized, which can be used to reconstruct light fields from arbitrary light field imaging systems, independent of whether the camera is based on microlenses, mirrors, or coded apertures, or whether it is realized by employing a camera array.

In the following section, related works in the field of light field decoding and reconstruction are presented. Then, in Sec. 6.2, a new generic approach for light field reconstruction is proposed that only uses the information contained in the set of rays obtained via the generic camera calibration. Finally, Sec. 6.3 experimentally validates the proposed method by reconstructing real light fields obtained with different light field acquisition systems and compares it to state-of-the-art methods.

6.1 Related Works

The first work on light field calibration was done in the context of multi-camera arrays [208]. However, these cannot simply be transferred to other light field acquisition systems such as MLA-based light field cameras. Due to their complex design, the light field has to be decoded from the raw sensor image using sophisticated algorithms. Furthermore, each lens (main lens and microlens) is affected by lens aberrations, *i.e.*, a subsequent rectification of the decoded light field is necessary to obtain correct geometric information relevant for image processing and optical metrology.

Among the microlens-based light field cameras, the standard plenoptic camera (or unfocused plenoptic camera) has been studied the most, as it is useful in consumer applications and image processing without requiring metric calibration [137]. To still be able to compensate for optical distortions, Ng and Hanrahan [136] suggested a digital correction of the lens aberrations without metric calibration by digitally re-sorting aberrated rays to where they should have terminated. The first metric calibration of a commercial light field camera was proposed by Dansereau *et al.* [44]. To decode the light field from the sensor data, they first estimate the grid parameters of the MLA. This is done by detecting the microlens centers from corresponding white images and building a regular grid that best approximates the detected centers. The light field is then decoded by assigning a spatial coordinate to each microlens and an angular coordinate to every pixel under each microlens, and by converting the hexagonal grid of microlenses into a rectangular one. Subsequently, the decoded light field is calibrated using a camera model consisting of ten intrinsic parameters and five distortion parameters, allowing the SAIs to be corrected by inverting the distortions. In this process, the calibration is initialized using the SAIs and then refined by minimizing the ray re-projection error, *i.e.*, the distance between the 3D positions of checkerboard features and the camera rays. Cho *et al.* [38] perform an erosion operation on the white image and estimate the microlens centers by using clustering and a parabolic fitting. They then decode the light field directly from the hexagonal layout using a barycentric interpolation. However, they neither perform metric calibration nor rectification. Bok *et al.* [20], in contrast, presented a method that can extract a rectified light field directly from raw sensor data, avoiding intermediate reconstruction steps.

In addition, they introduce a new projection model for microlens-based light field cameras that contains a smaller number of parameters than the previous methods. Instead of checkerboard corner features, they use line features extracted directly from the raw data. Further, the microlens centers are calculated individually without fitting a grid and the light field is decoded by barycentric interpolation. Eventually, the light field is rectified and the camera parameters are calculated by minimizing the distance between line segments and camera rays. Since all methods rely on a correct description of the microlens grid, Schambach and Puente León [180] propose an extended model that additionally takes into account the natural and mechanical vignetting of the microlenses and main lens. As a consequence, the calibration becomes more accurate, especially in SAIs corresponding to the peripheral regions of the angular dimension where the vignetting effect is more prominent.

For a focused plenoptic camera, the distance between the MLA and the sensor is not equal to the microlens' focal length. As a result, these cameras achieve a higher spatial resolution with decreasing angular resolution. To further increase the depth of field, the manufacturer Raytrix proposed multi-focus cameras in which the microlenses have different focal lengths [148]. Unlike the unfocused plenoptic camera, where each pixel under the microlens can be assigned to an SAI, the (multi-)focused plenoptic camera works like a micro camera array, where each microlens can be interpreted as a virtual camera observing a very small section of the scene. By using neighboring microlenses to perform stereo-based triangulation, a virtual depth map can be estimated. And by stitching the micro-images together using this depth information, an all-in-focus image of the scene can be reconstructed. However, because the virtual depth map can only be interpreted in a relative manner, a metric calibration is necessary. A first approach for the calibration of a multi-focus plenoptic camera was suggested by Johannsen *et al.* [97]. They extract a depth map and an all-in-focus image from the camera data and model the resulting synthetic image using a 15-parameter model that includes lateral distortion as well as a depth-dependent distortion. Heinze *et al.* [80] extended the model by considering the different focal lengths of the microlenses. Zeller *et al.* [239] introduced a new depth distortion model that is directly derived from the theory of depth estimation in a

focused plenoptic camera, and in addition, they extended the residual of their optimization to three dimensions by including the virtual depth. A disadvantage of the above methods is that they depend on Raytrix's software package since they do not start from raw data but the synthetic all-in-focus image and the virtual depth map.

6.2 Generic Light Field Reconstruction

To be able to extract light field information from the raw data, the previously discussed methods must initially detect the centers of the microlenses with high precision. But even with a subpixel accurate detection, most of the time only the rays near the center of the microlenses are precisely calibrated. The camera rays at the boundary of the microlenses are very difficult to model in all approaches, and therefore these pixels are often discarded. Another disadvantage of the classical methods is the model-based calibration in general. It cannot describe highly local errors such as the strong distortions at the boundaries of the microlenses using only a low-dimensional model. Hence, a generic camera calibration should be advantageous. However, the biggest disadvantage of the common light field reconstruction methods is that they each are only applicable to a single type of camera. For example, the methods by Dansereau *et al.* [44] and Bok *et al.* [20] can only be used with MLA-based light field cameras whose microlenses are exactly focused onto the sensor.

Since the calibrated rays describe the camera very well, it also makes sense to make use of it for the light field reconstruction. In fact, the generic ray bundle already represents the light field perfectly and optimally takes into account all distortions of the camera optics. More precisely, this means that the set of rays is effectively an irregularly sampled version of the distortion-free light field. For the light field reconstruction, this implies that no specific model of the used camera has to be developed, the sensor data does not have to be decoded according to this model, it is not necessary to detect the centers of any microlenses, and no hexagonal sampling of an MLA has to be compensated. Instead, the irregularly sampled light field has to be transformed into an adequate representation. Of course, since this is completely independent of the camera optics used,

a fully generic reconstruction algorithm is obtained that can be applied to any type of light field camera.

In the following, it will be explained how a conventional light field can be reconstructed from the unconstrained ray bundle. For this purpose, a regular and discrete parametrization of the target light field is first found based on the irregular data. Subsequently, it is shown how the irregular data is interpolated in a suitable way to this newly defined regular grid of light field pixels. And finally, to be useful for optical metrology applications, the intrinsic parameters of the reconstructed light field are derived.

6.2.1 Parametrization of Light Field Coordinates

To decode a light field from the raw sensor data, the camera must first be calibrated, *e.g.*, by using a generic calibration method as described in Sec. 5.2. As a consequence, all the preprocessing steps of the conventional state-of-the-art light field calibration algorithms are not needed at all. Even more, it does not actually matter what type of light field acquisition device is used. After applying the generic camera calibration, a ray bundle is obtained in an arbitrary coordinate system, which can easily be transformed into a camera-fixed coordinate system using the normalization presented in Sec. 5.2.7. Since most light field algorithms do not work with Plücker-coordinates, as the last step, the camera ray parameters are transformed into light field coordinates. To do so, the rays are first transformed into the camera-fixed coordinate system, by shifting the origin and rotating the axes. Afterward, the intersections of the rays with the two-plane representation of the light field are calculated. For this, the u, v -plane is placed orthogonal to the z -axis into the origin of the coordinate system, *i.e.*, this corresponds approximately to the center of the camera's exit pupil when an MLA-based light field camera is used. The s, t -plane is placed parallel to this at an arbitrary distance f , see figure 6.1. Thus, each camera ray $\mathbf{l}_i = (\mathbf{d}_i, \mathbf{m}_i)^T$ can be described by four light field coordinates $\bar{s}_i, \bar{t}_i, \bar{u}_i, \bar{v}_i$:

$$\lambda (\bar{s}_i, \bar{t}_i, \bar{u}_i, \bar{v}_i, 1)^T = \mathbf{P} \mathbf{T} \mathbf{l}_i, \quad (6.1)$$

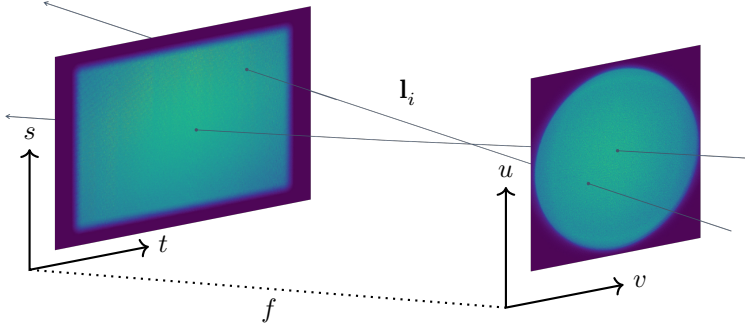


Figure 6.1 Two-plane parametrization of the light field. The ray l_i intersects the u, v - and the s, t -plane in (s_i, t_i, u_i, v_i) . The intensities in the planes visualize the spatial distribution of the intersection points as a 2D histogram. The u, v -plane lies in the plane of the camera's main lens. The s, t -plane corresponds to a projection on the rectangular sensor.

with $\lambda \neq 0$, using the coordinate transformation matrix \mathbf{T} that is derived in Sec. 5.2.7, and with a ray-to-light-field projection operator \mathbf{P} [94]:

$$\mathbf{P} = \begin{pmatrix} f & 0 & 0 & 0 & -1 & 0 \\ 0 & f & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}. \quad (6.2)$$

6.2.1.1 Regular Light Field Grid

To reconstruct a light field from the bundle of rays associated with the camera, the calibrated ray coordinates must first be transformed into a standardized grid. Afterwards, the observed ray intensities $L(\bar{s}_i, \bar{t}_i, \bar{u}_i, \bar{v}_i)$ can be interpolated to a discretized light field, which is parametrized in the same two-plane representation as previously. The complete set of real camera rays, which is described as a set of 4D points, is arranged in an irregular 4D grid. Still, the classical light field algorithms (*e.g.*, refocusing and depth estimation) require a regular grid with uniform spacing. Therefore, this irregular grid of continuous ray coordinates has to be interpolated to a discrete light field described by a regular grid.

Hence, it is necessary to define a regular grid with integer grid points

$$(s, t, u, v) \in [0, N_s - 1] \times [0, N_t - 1] \times [0, N_u - 1] \times [0, N_v - 1] \quad (6.3)$$

with a fixed number of samples N_s, N_t, N_u, N_v in the respective dimensions. After the discrete target light field has been defined, the set of real camera rays are transformed, for which the parameter space of the actual ray geometry must be estimated. For this, the domains of the real light field dimensions have to be determined by analyzing the intersection points of the rays with both planes of the light field representation. It goes without saying that among all rays there are also isolated outliers that deviate so strongly from the others that it is not worthwhile to consider them in the interpolation. Therefore, the 2D densities of the intersection points should be investigated by making use of a 2D histogram analysis, see figure 6.1. To place the regular grid structure into the 2D density of the irregular data, a threshold value on the histogram data enables defining the grid extension. A threshold of, *e.g.*, 1% ensures that most of the camera's rays are within the range defined by the grid.

Since the real light field parameters are specified in physical units, *e.g.*, millimeter, they have to be transformed to the previously defined discrete 4D-pixel grid by shifting the minimal value s_o, t_o, u_o, v_o , normalizing the width of the histogram $\Delta s, \Delta t, \Delta u, \Delta v$, and considering the number of samples. The normalized coordinates are then defined by

$$\begin{aligned} s_i &= \frac{\bar{s}_i - s_o}{\Delta s} (N_s - 1), & t_i &= \frac{\bar{t}_i - t_o}{\Delta t} (N_t - 1), \\ u_i &= \frac{\bar{u}_i - u_o}{\Delta u} (N_u - 1), & v_i &= \frac{\bar{v}_i - v_o}{\Delta v} (N_v - 1). \end{aligned} \quad (6.4)$$

This still results in irregularly spaced data, which however can now be interpolated more easily to obtain the desired regularly sampled light field. The number of 4D cubes in each direction and the length of their edges could in principle be defined arbitrarily, but it is advisable to incorporate knowledge about the physical camera. For example, the Lytro Illum camera considered in this work has microlenses with a diameter of about 15 pixels. Thus, because the camera is of the unfocused design, this sampling can be used directly as a basis for the discretization of the angular coordinates of the u, v -plane, where $N_u = N_v \approx 15$ due to the

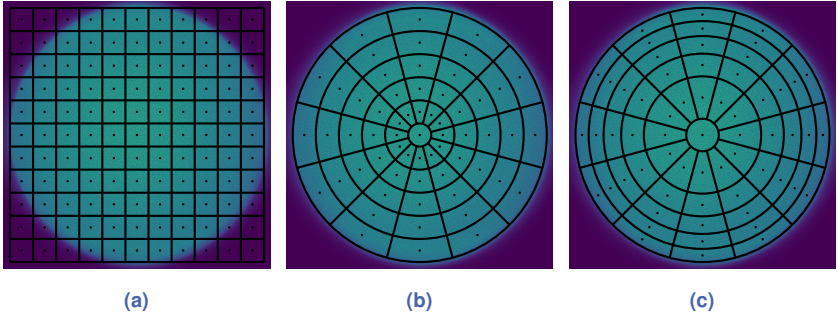


Figure 6.2 Different sampling patterns of the u,v -plane: The dots represent the pixel coordinate, and the lines limit the pixel area. (a) Cartesian sampling. (b) Polar sampling with equidistant radius spacing, using $R_n = \frac{1}{2} + n$. (c) Polar sampling with equal pixel area, using $R_n \sim \sqrt{nN_\phi + 1}$.

circular shape of the camera's main lens. The sampling of the s, t -plane can be determined in the same way, e.g., by the number of microlenses in front of the sensor, whereby it is advisable to choose $\frac{N_t-1}{\Delta t} = \frac{N_s-1}{\Delta s}$ to obtain square-shaped spatial pixels.

6.2.1.2 Polar Parametrization of Angular Coordinates

As can be seen in figure 6.2, the parametrization of the u, v -plane using Cartesian coordinates is not always ideal. If the grid is defined to enclose the entire circle, then the light field is reconstructed in areas where no rays pass through the u, v -plane. If the grid is placed inside the circle, a sufficient number of rays will pass through each light field pixel. However, information is discarded at the edges. Hence, it would be better to directly use a polar parametrization of the angular coordinates, which would allow the entire information to be captured without sampling unneeded areas. Therefore, the angular coordinates are defined by polar coordinates r and ϕ . To further obtain a resolution comparable to the Cartesian sampling, the number of samples is chosen to be $N_r = N_u$ and $N_\phi \approx N_u$. The coordinates are then linearly sampled in the domain $r \in \left[-\frac{N_r-1}{2}, \frac{N_r-1}{2}\right]$ and $\phi \in \left[0, \frac{N_\phi-1}{N_\phi}\pi\right]$. Here, N_ϕ should be a multiple of 4 to be able to obtain horizontal and vertical EPIs com-

parable to the Cartesian sampling, *i.e.*, $\text{EPI}(r, s)|_{\phi=0} = \text{EPI}(u, s)$ and $\text{EPI}(r, t)|_{\phi=\frac{\pi}{2}} = \text{EPI}(v, t)$. While the advantage of a polar parametrization is a more efficient sampling, there are also disadvantages. When sampling the angle and the radius in equidistant steps, the effective pixel size grows with increasing radius, see figure 6.2(b). As a result, fewer rays pass through smaller pixels, which would result in a lower signal-to-noise ratio for these pixels during the interpolation to the discrete light field.

A possible solution here is to define the radius sampling in such a way that each pixel has the same area. This can easily be achieved by not using a linear sampling of the radius coordinates but by transforming their domain, see figure 6.2(c). The radius R_0 of the center-most pixel has the area A_0 , whereas the area A_n of the remaining pixels is represented by a sector of an annulus:

$$A_0 = \pi R_0^2, \quad (6.5)$$

$$A_n = \pi(R_n^2 - R_{n-1}^2) \frac{1}{N_\phi} \text{ for } n > 0. \quad (6.6)$$

By requiring $A_n = A_{n-1} = \dots = A_0$ and using mathematical induction, a formula for the radius is obtained:

$$R_n = R_0 \sqrt{n N_\phi + 1} = \frac{N_r}{2} \sqrt{\frac{n N_\phi + 1}{\frac{N_r-1}{2} N_\phi + 1}}. \quad (6.7)$$

The reconstruction of the light field using polar coordinates can then be performed in the same way as when Cartesian coordinates are used. The only distinction is the different sampling grid in the angular plane, for which the Cartesian coordinates u_i, v_i need to be transformed into polar coordinates r_i, ϕ_i using

$$r_i = \text{sign}(v_i) \sqrt{u_i^2 + v_i^2}, \quad (6.8)$$

$$\phi_i = \arctan2\left(v_i - \frac{N_v - 1}{2}, u_i - \frac{N_u - 1}{2}\right) \bmod \pi. \quad (6.9)$$

A reverse transformation from polar coordinates back to Cartesian coordinates is achieved by

$$u_i = r_i \cos \varphi_i + \frac{N_u - 1}{2}, \quad (6.10)$$

$$v_i = r_i \sin \varphi_i + \frac{N_v - 1}{2}. \quad (6.11)$$

The only differences between polar sampling with equidistant radius spacing and polar sampling with equal pixel area are that the discrete pixel coordinates are slightly different. However, using a polar parametrization for conventional light field applications could be a challenge, since existing algorithms are based on rectangular data. In particular, commonly used techniques based on CNNs cannot work with this representation without further modification, since the standard convolution operators would first have to be replaced by polar ones.

6.2.2 Weighted Interpolation of Irregular Data

After the parameters of the light field have been defined, each corresponding light field pixel can be determined for every ray by finding the discrete grid point that is closest to the ray's light field coordinates. Since the rays and the grid are normalized to the same scale, the set of rays $\mathcal{N}_{s,t,u,v}^m$ that affects a pixel (s, t, u, v) can be found using a rounding operation to the closest integer $\lceil \cdot \rceil$. As a result, each light field pixel is only influenced by rays that lie in the corresponding 4D cube

$$\mathcal{N}_{s,t,u,v}^m := \left\{ i : \frac{m}{2} \geq \left\| \begin{pmatrix} x \\ y \\ u \\ v \end{pmatrix} - \begin{pmatrix} \lceil x_i \rceil \\ \lceil y_i \rceil \\ \lceil u_i \rceil \\ \lceil v_i \rceil \end{pmatrix} \right\|_\infty \right\}, \quad (6.12)$$

where each individual ray is assigned to the nearest pixel when using $m = 1$. When using a polar parametrization of the angular coordinates, the parameters u, v need to be replaced by r, ϕ . To allow a ray to influence more than the nearest pixel, higher-order neighbors can be utilized with $m > 1$, $m \in \mathbb{N}^+$. The intensity of a discrete pixel can then be calculated

from the intensity values of the corresponding rays as a weighted average:

$$L(s, t, u, v) = \frac{\sum_{i \in \mathcal{N}_{s,t,u,v}^m} w_i(u, v, s, t) L(s_i, t_i, u_i, v_i)}{\sum_{i \in \mathcal{N}_{s,t,u,v}^m} w_i(u, v, s, t)}. \quad (6.13)$$

For the weighting factor, the distance between a ray's light field parameters and its correspondence in the grid is calculated. In order to consider larger deviations less, the error is squared and exponentially weighted:

$$w_i(u, v, s, t) = \frac{1}{\varepsilon_i} \exp \left(-\|(s, t, u, v)^T - (s_i, t_i, u_i, v_i)^T\|^2 \right). \quad (6.14)$$

A separate weighting of the individual light field coordinates is not required because these have already been brought to a unified basis by the normalization (6.4). To additionally benefit from the results of the generic camera calibration, an error measure ε_i is taken into account, *e.g.*, the pixel-wise ray re-projection error (5.34). This suppresses poorly calibrated camera rays, which often do not have good optical properties, *e.g.*, dead pixels or pixels at the edges of microlenses, which can be strongly distorted.

Regarding computational resources, it remains to say that the direct calculation of the set of nearest neighbors $\mathcal{N}_{s,t,u,v}^m$ is at first extremely inefficient. Since for each discrete pixel (s, t, u, v) a complete search over all irregularly distributed rays (s_i, t_i, u_i, v_i) must be performed, the complexity is $\mathcal{O}(n^2)$, with n being the number of pixels. Using more efficient algorithms, such as k-d trees, can decrease the complexity to $\mathcal{O}(n \log n)$ [39]. Fortunately, however, due to the ray coordinates being normalized to a convenient range, it is even better to simply assign each irregular coordinate (s_i, t_i, u_i, v_i) to a discrete pixel directly. This is much faster since the nearest neighbor of a continuous coordinate is directly its closest integer analogon. Hence, a rounding of the ray coordinates directly returns the corresponding set of nearest neighbors. In addition, to allow the assignment of higher-order neighbors, a formula can be given that allocates the rays to a set $\mathcal{N}_{s,t,u,v}^m$ using only fast and simple operations:

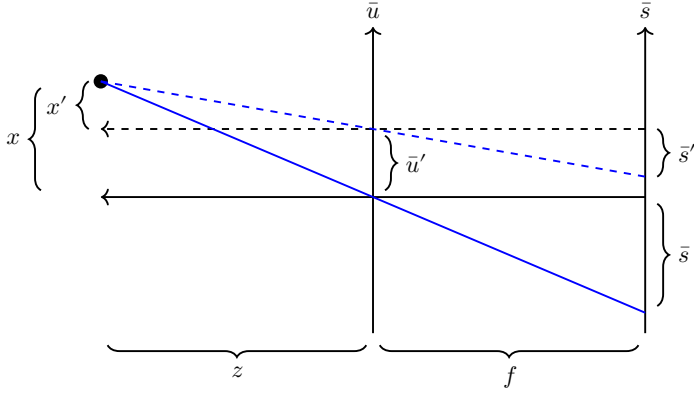


Figure 6.3 Perspective projection through shifted pinhole onto shifted sensor plane.

$$\begin{pmatrix} u \\ v \\ s \\ t \end{pmatrix} \equiv \left[\begin{pmatrix} \lceil u_i \rceil \\ \lceil v_i \rceil \\ \lceil s_i \rceil \\ \lceil t_i \rceil \end{pmatrix} + \frac{m-1}{2}(-1)^m \begin{pmatrix} \text{sign}(u_i - \lceil u_i \rceil) \\ \text{sign}(v_i - \lceil v_i \rceil) \\ \text{sign}(s_i - \lceil s_i \rceil) \\ \text{sign}(t_i - \lceil t_i \rceil) \end{pmatrix} \right] \Rightarrow i \in \mathcal{N}_{s,t,u,v}^m \quad (6.15)$$

Hence, a complexity of $\mathcal{O}(n)$ is achieved. Even more, due to all rays being independent of one another, the creation of the nearest neighbor set (6.12) and the weighted interpolation (6.13) can be parallelized using GPU hardware. The reconstruction of a complete light field then takes only a few seconds (in the case of the Lytro Illum with a 40 Mpx Sensor, using an Nvidia GTX 1080 Ti, an Intel Core i7-6700, and 16 GB RAM).

6.2.3 Intrinsic Camera Parameters

Apart from the radiometric reconstruction of the light field, the geometric ray properties are relevant in many applications. For optical metrology, 3D reconstruction, or other areas of computer vision, a mapping is needed to transform pixel coordinates into world coordinates and, vice versa, to project points from world coordinates onto the pixel plane. Hence, to use the light field camera for the deflectometric reconstruction of specular

surfaces, as will be presented in Ch. 7, the intrinsic camera parameters need to be available. Unlike the classic pinhole camera model, where each world point is mapped to only a 2D pixel pair, the same world point can be mapped to more than one 4D light field pixel. Illustratively, this can be understood by the observation that a light field camera can also be interpreted as an array of individual virtual sub-cameras, where an observed point is mapped to a 2D pixel pair in each individual camera's virtual sensor plane. Hence, every angular coordinate needs a projection equation from world points to the respective spatial pixels. The perspective projection of a point $\mathbf{x} = (x, y, z)^T$ onto spatial pixels through a pinhole located at the optical center is obtained with (5.1). Following figure 6.3, projecting the same point through a pinhole located at the shifted position \bar{u} then results in

$$\bar{s}' = \frac{fx'}{z}, \text{ with } \bar{s}' = \bar{s} - \bar{u}, x' = x - \bar{u}, \quad (6.16)$$

$$\Rightarrow \bar{s} = \frac{f(x - \bar{u})}{z} + \bar{u}. \quad (6.17)$$

In conclusion, the intrinsic camera parameters required for the perspective projection of a point \mathbf{x} onto the spatial pixels $(\bar{s}, \bar{t})^T$ are described for each angular coordinate \bar{u}, \bar{v} by a projection matrix (comparable to the standard pinhole camera model from Sec. 5.1). Since the optical centers of the individual sub-cameras are slightly displaced to each other in the \bar{u}, \bar{v} -plane, a corresponding translation vector is required to represent the relative offset to the central sub-camera. The projection is represented by

$$\begin{pmatrix} \bar{s} \\ \bar{t} \\ 1 \end{pmatrix} \sim \mathbf{K}(\mathbf{x} + \mathbf{t}), \text{ with } \mathbf{K} = \begin{pmatrix} f & 0 & \bar{u} \\ 0 & f & \bar{v} \\ 0 & 0 & 1 \end{pmatrix}, \mathbf{t} = \begin{pmatrix} -\bar{u} \\ -\bar{v} \\ 0 \end{pmatrix}. \quad (6.18)$$

For every SAI, the pinhole is shifted in \bar{u}, \bar{v} -direction, and the respective center of the sensor is shifted in the opposite direction. Since all SAIs share the same virtual sensor plane, such parametrization results in an interesting effect that shows up in many light field camera calibration algorithms: negative disparities can be obtained. For camera array-based light field cameras, the minimal disparity is usually zero and corresponds to a point at optical infinity. In contrast, the plane of zero disparity in the

configuration presented here is the “focal plane” of the light field, where $z = f$. Points closer to the camera have a positive and points farther away a negative disparity. Hence, a point at the focal plane is imaged to the same spatial coordinate $\bar{s} = x$ in every sub-aperture image. Of course, the choice of the focal length f influences the light field representation. However, the actual value is not important, since it only results in a different light field parametrization. Also, the “focal plane” of the light field should not be confused with the plane where the imaging has the highest sharpness. Theoretically, a light field camera has an infinite depth of field. Practically however, due to the SAIs of a real light field camera being imaged through only a very small aperture, their depth of field is finite but sufficiently high [60]. In summary, the value of f does not correspond to a conventional focus.

Because the light field reconstruction is performed with regular grid parameters and discrete pixels (s, t, u, v) , the corresponding intrinsic parameters need to be derived to project world points to pixel coordinates. Thus, this results for every SAI in a camera matrix and a translation vector

$$\mathbf{K}_{uv} = \begin{pmatrix} f_s & 0 & c_s(u) \\ 0 & f_t & c_t(v) \\ 0 & 0 & 1 \end{pmatrix}, \mathbf{t}_{uv} = \begin{pmatrix} t_s(u) \\ t_t(v) \\ 0 \end{pmatrix}. \quad (6.19)$$

The corresponding parameters can directly be determined from the two-plane parametrization of the light field by using (6.4) and (6.17):

$$f_s = f \frac{N_s - 1}{\Delta s}, \quad (6.20)$$

$$f_t = f \frac{N_t - 1}{\Delta t}, \quad (6.21)$$

$$c_s(u) = u \frac{\Delta u}{\Delta s} \frac{N_s - 1}{N_u - 1} + (N_s - 1) \frac{u_o - s_o}{\Delta s}, \quad (6.22)$$

$$c_t(v) = v \frac{\Delta v}{\Delta t} \frac{N_t - 1}{N_v - 1} + (N_t - 1) \frac{v_o - t_o}{\Delta t}, \quad (6.23)$$

$$t_s(u) = -u \frac{\Delta u}{N_u - 1} - u_o, \quad (6.24)$$

$$t_t(v) = -v \frac{\Delta v}{N_v - 1} - v_o. \quad (6.25)$$

In this context, when using a polar parametrization of the angular coordinates, the intrinsic parameters can be calculated for each r, ϕ -pair by transforming them to their corresponding u, v -value using (6.10) & (6.11).

The forward projection of a point $\mathbf{x} = (x, y, z)^T$ (measured in the coordinate system fixed to the central subcamera) onto the light field pixel (s, t, u, v) can be found with

$$z \begin{pmatrix} s \\ t \\ 1 \end{pmatrix} = \mathbf{K}_{uv} (\mathbf{x} + \mathbf{t}_{uv}). \quad (6.26)$$

The backward projection of light field pixels (s, t, u, v) to points $\mathbf{x}(z)$ along the associated camera ray is given by

$$\mathbf{x}(z) = z \mathbf{K}_{uv}^{-1} \begin{pmatrix} s \\ t \\ 1 \end{pmatrix} - \mathbf{t}_{uv}. \quad (6.27)$$

Finally, for every light field coordinate (s, t, u, v) , the corresponding ray in Plücker-coordinates can be obtained easily as

$$\mathbf{d}(s, t, u, v) = \frac{\mathbf{x}(z) - \mathbf{x}(0)}{\|\mathbf{x}(z) - \mathbf{x}(0)\|}, \quad (6.28)$$

$$\mathbf{m}(s, t, u, v) = \mathbf{x}(0) \times \mathbf{d}(s, t, u, v). \quad (6.29)$$

When the light field camera is used for depth estimation, the disparity of a scene feature is estimated [218]. The disparity appears as the slope of a line in the EPIs, as detailed in Sec. 3.2. Because negative disparities may also be observed with the light field parametrization presented here, the SAIs must first be brought to a uniform basis, *i.e.*, a disparity offset must be subtracted. Using the disparity and with the help of the baseline between the SAIs, it is then possible to convert back to the metric depth:

$$z = \frac{f_s b_s}{d_s - d_{\text{offset},s}} = \frac{f_t b_t}{d_t - d_{\text{offset},t}}, \quad (6.30)$$

where d_s, d_t represent the disparity estimated from the horizontal and vertical EPI, b_s and b_t represent the baselines in the respective directions,

and $d_{\text{offset},s}$ and $d_{\text{offset},t}$ are the offsets of the disparity. They can be calculated from the intrinsic parameters:

$$b_s = t_s(0) - t_s(1), \quad d_{\text{offset},s} = c_s(0) - c_s(1), \quad (6.31)$$

$$b_t = t_t(0) - t_t(1), \quad d_{\text{offset},t} = c_t(0) - c_t(1). \quad (6.32)$$

If, in addition, the light field resolution is chosen to result in square-shaped spatial and angular pixels $\Delta s(N_t - 1) = \Delta t(N_s - 1)$ and $\Delta v(N_u - 1) = \Delta u(N_v - 1)$, then it follows that $b_s = b_t$ and $d_{\text{offset},s} = d_{\text{offset},t}$, which means that the disparities calculated from different EPs should be equal.

6.3 Evaluation

This section evaluates the presented light field reconstruction algorithm and compares it to the state of the art. To show the advantage of the presented generic approach, three different light field camera systems are evaluated: a Lytro Illum with an RGB sensor, a monochromatic Raytrix R5, and a prototype K|Lens lens mounted onto an RGB camera sensor. The Lytro Illum and the Raytrix R5 are both microlens-based light field cameras. The former is an unfocused plenoptic camera [137], and the latter is a focused plenoptic camera [123], see Sec. 3.2. The K|Lens camera is based on an “Image Multiplier”, which contains a mirror tunnel, similar to a kaleidoscope. Using this, a multi-view capture of the scene is directly generated and mapped onto the camera sensor [128]. Consequently, all three cameras are based on very different camera models, and for conventional camera calibration, all would need a different calibration procedure. However, a generic calibration works independently of the camera. Here, the ray geometry of the vision rays of each camera was estimated using the generic camera calibration from Sec. 5.2.1. Subsequently, test scenes were captured to be used as a basis for the comparison of the proposed light field reconstruction.

In the following, first, the presented algorithm is analyzed, and a qualitative evaluation of the generic light field reconstruction for all light field cameras is conducted. Then, the quality of the geometric reconstruction is investigated by a quantitative comparison of the calibration error.

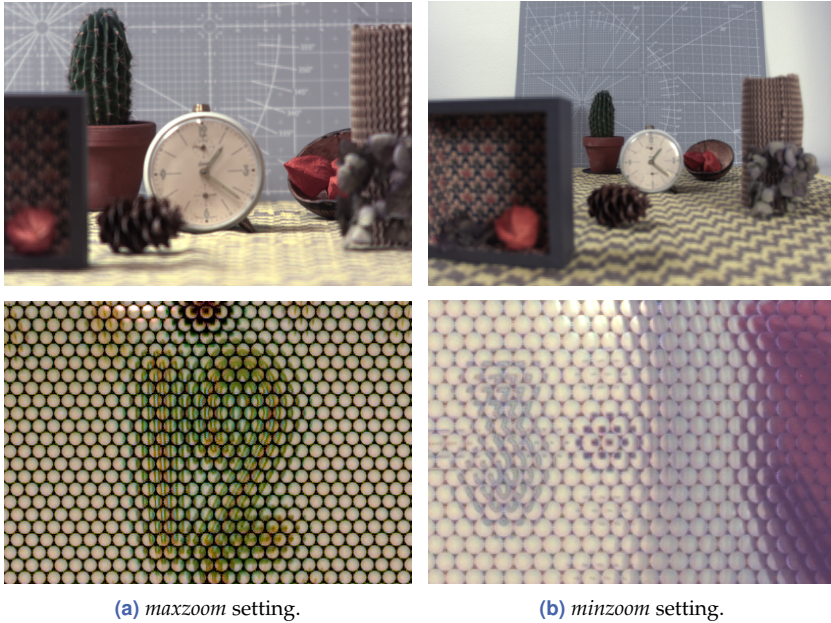


Figure 6.4 Lytro Illum data. Top: Sensor image. Bottom: Detailed views.

6.3.1 Light Field Reconstruction: Lytro Illum

The Lytro Illum light field camera has a sensor of size 7728×5368 px with a pixel pitch of $1.4 \mu\text{m}$ overlaid with a Bayer pattern. Hence, with the help of demosaicing, color information can be obtained. In front of the sensor is an array of hexagonally arranged microlenses, with each microlens having an approximate diameter of $20 \mu\text{m}$ and a focal length of $f = 40 \mu\text{m}$. Since the camera is an unfocused plenoptic camera, the distance of the microlenses to the sensor plane corresponds to their focal length. The main lens of the camera is a zoom lens with a selectable focal length equivalent in the range of 30 mm to 250 mm. Therefore, two configurations are investigated for the Lytro Illum camera: A *maxzoom* setting with a focal length equivalent of 250 mm, and a *minzoom* setting with a focal length equivalent of 30 mm.

Figure 6.4 shows the sensor data corresponding to both zoom settings after demosaicing. From a coarse point of view, the images look like the

image using a conventional camera. Only when taking a closer look the microlenses can be seen. It can be seen that the f-number matching does not work perfectly for the *maxzoom* setting since the microlenses show strong vignetting effects here. For the *minzoom* setting, these effects also occur, but not quite as strongly. To compensate for vignetting, the images are divided by a so-called white image, *i.e.*, an image of a white scene taken with the aid of an optical diffuser. The pre-processed raw data can then be used in the light field reconstruction algorithms. For a comparison of the presented generic method to the state of the art, the light field reconstruction methods of Dansereau *et al.* [44] and Bok *et al.* [20] are evaluated as well. Both methods only work with unfocused plenoptic cameras and can thus only be tested on the Lytro Illum data sets.

6.3.1.1 Ray distribution and Grid Parameters

With a pixel pitch of $1.4\text{ }\mu\text{m}$ and a microlens diameter of $20\text{ }\mu\text{m}$ there are about 14.3×14.3 pixels underneath each microlens. Since the Lytro Illum is of unfocused design, this corresponds directly to the angular resolution. A discrete angular sampling can then be found by rounding up or down. To obtain a central SAI, the angular resolution should be an odd number. Following Sec. 6.2.1, the spatial resolution can be found by dividing the sensor size by this angular sampling factor, resulting in a spatial resolution of approximately 520×376 px for each SAI. Still, to allow for a meaningful discussion of the proposed light field reconstruction relative to other methods in the literature, the Lytro Illum data is evaluated by choosing the resolution of the light field grid to be $(N_s, N_t, N_u, N_v) = (625, 434, 15, 15)$, which is the same as the reconstructed light field of Dansereau *et al.* In comparison, the light field obtained from the reconstruction method by Bok *et al.* has a resolution of $(N_s, N_t, N_u, N_v) = (552, 383, 13, 13)$, meaning that the worst pixels at the edges of the microlenses are cut off.

Now that the resolution of the discrete target light field is known, the real light field parameters must be transformed into this newly defined 4D grid. For this purpose, the intersections of the vision rays with the two planes of the two-plane parametrization of the light field are analyzed. The histogram analysis of the intersection points for the *minzoom* and *maxzoom* setting are shown in figure 6.5. For the *maxzoom* setting, the

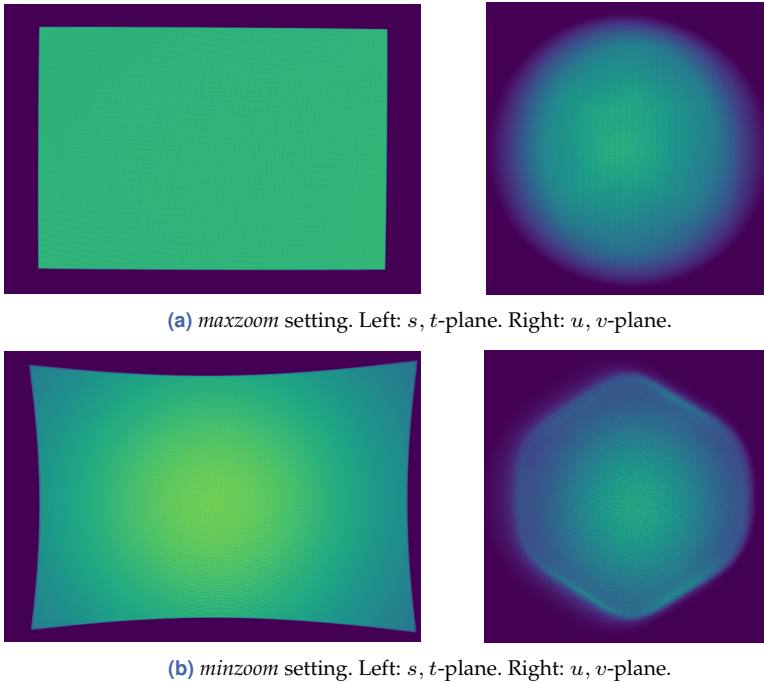


Figure 6.5 Lytro Illum: Histogram of ray-plane intersections.

histograms are very regular. The u, v -plane shows a circular distribution. Since the u, v -plane is placed at the point of the highest ray density, one can indirectly observe the aperture of the main lens here. The diameter of the aperture is estimated to be 3.4 cm, which also corresponds approximately to what can be roughly measured with a tape measure when looking into the objective from outside the camera. The s, t -plane shows a rectangle, which corresponds to a projection of the rectangular sensor. The extension of this rectangle depends on the arbitrarily chosen distance f between the two planes and is therefore not important. The histograms of the *minzoom* setting show strong optical distortions. The s, t -plane shows a rectangle, which has a pincushion distortion. This precisely corresponds to the distortions produced by the non-ideality of the main lens. Since the generic camera model works completely independent of any

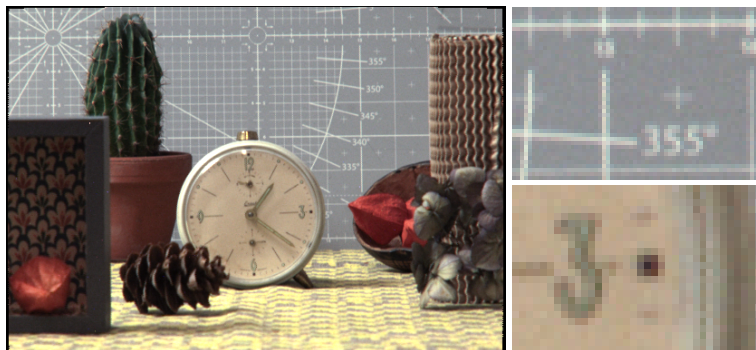
low-dimensional parametric model, this optical distortion is perfectly described by the generic set of vision rays. Interestingly, the u, v -plane is no longer circular but has a hexagonal structure. The aperture of the Lytro is therefore most likely hexagonal, meaning the projection of the aperture can be seen here. For the *maxzoom* setting, where the aperture is already very small, as seen in the white images, the aperture is probably “completely open” and therefore circular. To keep the f-number matching, the Lytro Illum seems to have a variable input aperture.

With the help of the histograms, the dimension of the grid parameters can be determined. For this purpose, rectangles are fitted around the histograms such that at least 99 % of the ray-plane intersections should lie within the rectangular area. This effectively suppresses outliers. Everything that is not exactly in the grid is not necessarily completely lost and can still have an influence on the neighboring light field pixels, as long as the number of nearest neighbors in (6.12) is chosen to be high enough.

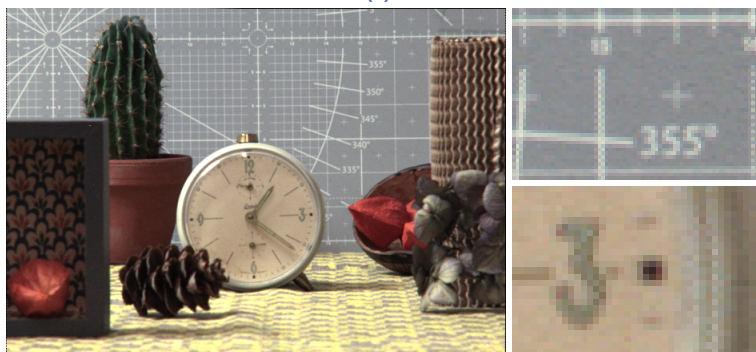
6.3.1.2 Qualitative Evaluation of Subaperture Images

Due to the relatively freely chosen sampling grid, in some cases no corresponding ray can be assigned for some of the discrete 4D pixels. For this reason, if the interpolation order is too low, this can lead to a perforated reconstruction. Hence, for the generic reconstruction, up to second-order-nearest neighbors were used for the angular domain by setting $m = 2$, and up to third-order-nearest neighbors were used for the spatial domain with $m = 3$ in (6.12). Increasing the order of interpolation too much does not change the result of the reconstruction significantly, because the exponential weight of (6.14) automatically punishes rays that are too far away very strongly. The only major disadvantage of a higher interpolation order is the longer reconstruction time, since the intensity of each ray must be considered for more than just the nearest neighbor.

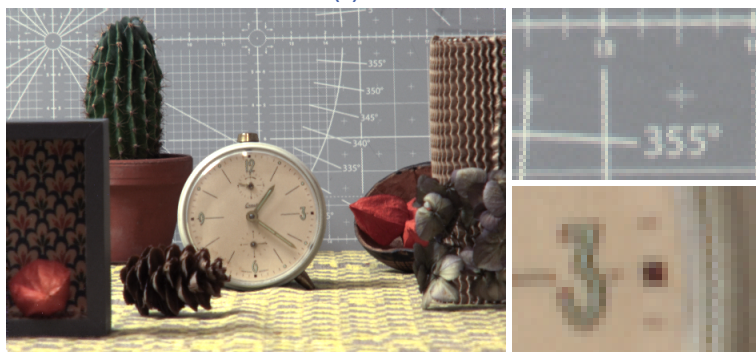
The reconstruction of the central SAI of the *maxzoom* dataset captured with the Lytro Illum is shown in figure 6.6. Here, only rays from the center of the u, v -plane were used in the reconstruction. It can be seen that the presented generic method can reconstruct the scene correctly, although there were absolutely no presumptions about the internal optical structure of the camera and no information on the correlations between rays and pixels on the sensor was used. In detail, it can be seen that the generic



(a) Bok *et al.*

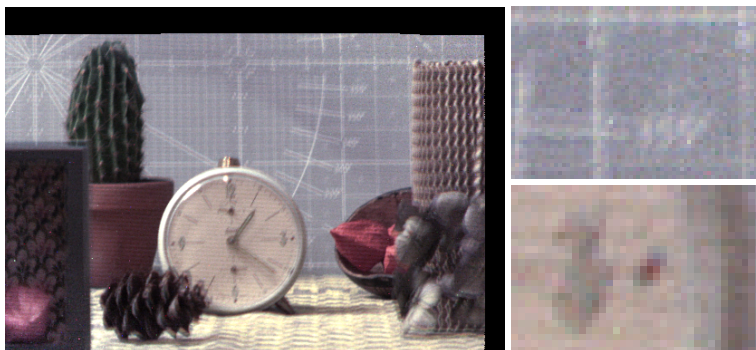
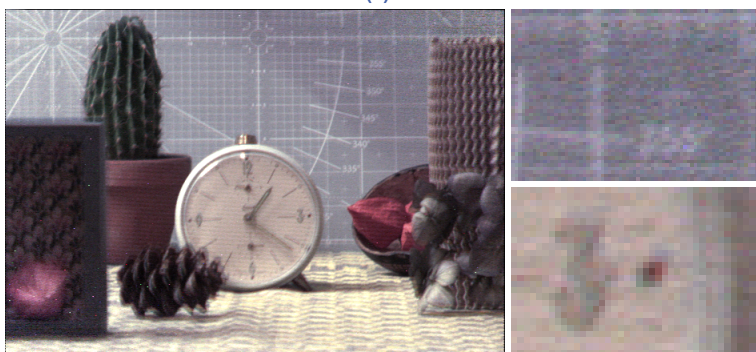
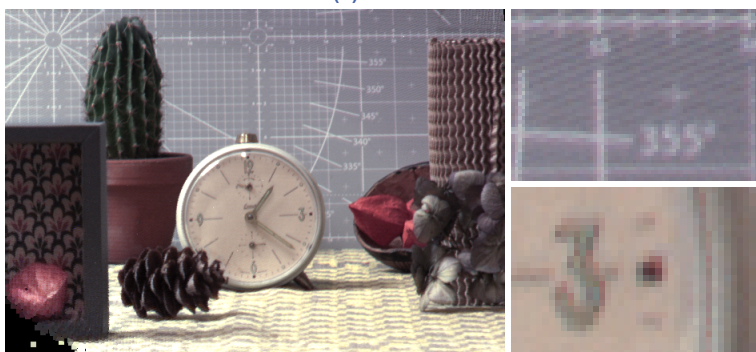


(b) Dansereau *et al.*



(c) Proposed generic light field reconstruction.

Figure 6.6 *maxzoom* setting. Left: SAI from the center of the u, v -plane. Right: Details.

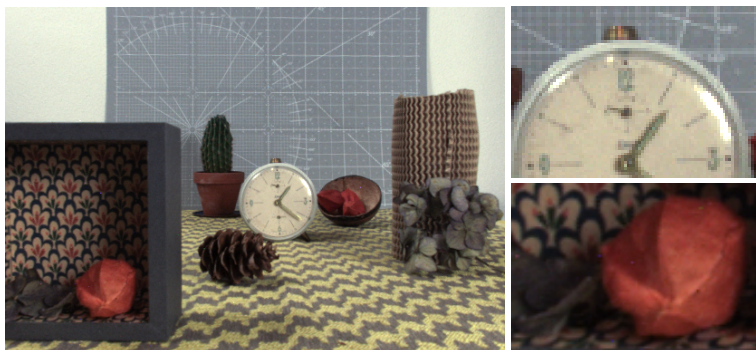
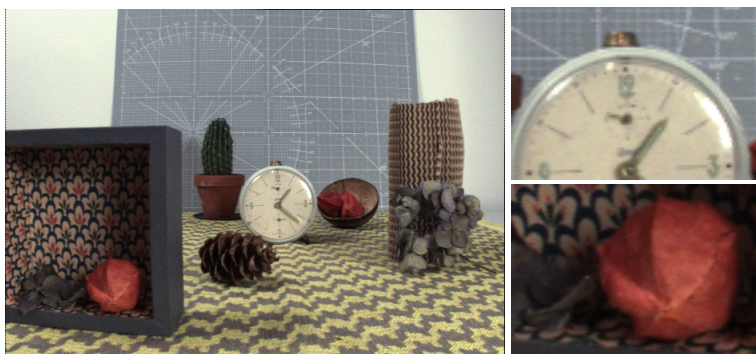
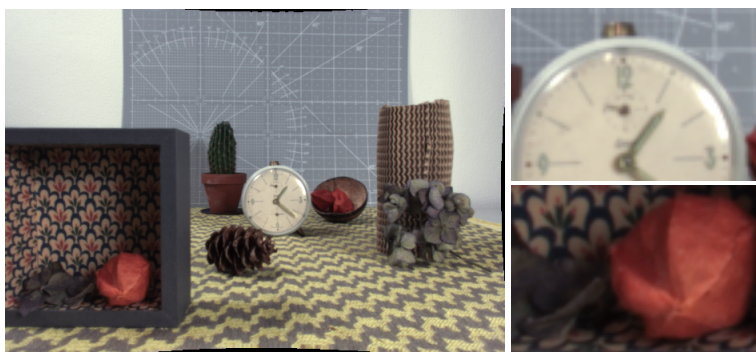
(a) Bok *et al.*(b) Dansereau *et al.*

(c) Proposed generic light field reconstruction.

Figure 6.7 *maxzoom* setting. Left: SAI from the edge of the u , v -plane. Right: Details.

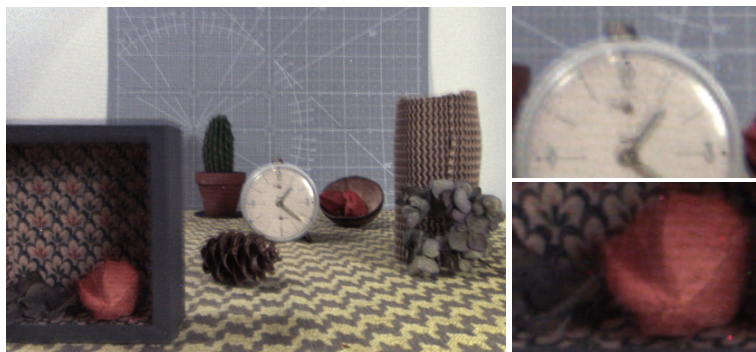
method can reconstruct the light field even near object edges very well. The reconstruction results of Dansereau *et al.* and of the generic method are relatively similar and show a slightly sharper result compared to the method of Bok *et al.* However, moving away from the center and looking at peripheral SAIs that still contain information, one sees that the quality of the images for Dansereau *et al.* and Bok *et al.* decreases significantly, while the result of the generic method only becomes slightly blurrier, see figure 6.7. In addition, the image of Bok *et al.* shows black borders, *i.e.*, invalidated pixels, at the top and on the right. The generic method shows a similar effect, which, depending on how tight the dimension of the s, t -plane is chosen using the histogram, could also be stronger. These pixels define areas of the light field where there is no real ray. Therefore, no information can be obtained. Bok *et al.* avoid this problem in the lower-left area by simply reducing the size of the image. For Dansereau *et al.*, a similar effect shows up, if one chooses SAIs that lie even further at the edge. However, since here their reconstruction is of such poor quality, it does not make sense to use it for the comparison made here.

The reconstruction of the central SAI of the *minzoom* dataset is shown in figure 6.8. Here one can see similar results to before. Dansereau *et al.*'s method shows the sharpest reconstruction followed by Bok *et al.*'s method. At the edge of the image, the generic reconstruction shows a similar performance to Bok *et al.*, visible in the bottom detail image. The minimally blurrier appearance of the generic reconstruction in the center near the alarm clock is due to the relatively freely chosen sampling of the light field. In order to reconstruct the entire light field, regions at the periphery of the image were also reconstructed in this case. And because the light field was strongly rectified, the area in the center of the image shrinks. Consequently, fewer pixels remain for this area. Bok *et al.* avoid this problem by heavily cropping the entire image. Dansereau *et al.* do not have this problem either, as they do not perform rectification and undistortion. Their rectification algorithm only works for the older Lytro camera, which has a relatively simple optical setup. But it does not yield useful results for the newer Lytro Illum, which has a more sophisticated lens setup that reduces optical aberrations and that enables a variable zoom setting. Eventually, this means that the light field camera model of Dansereau *et al.* is not generalizable, and it does not even seem to be ap-

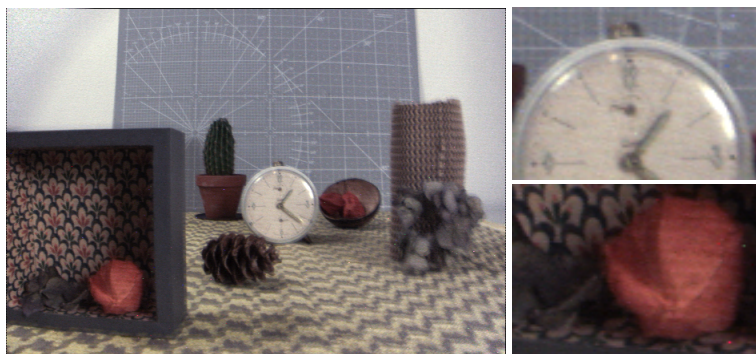
(a) Bok *et al.*(b) Dansereau *et al.*

(c) Proposed generic light field reconstruction.

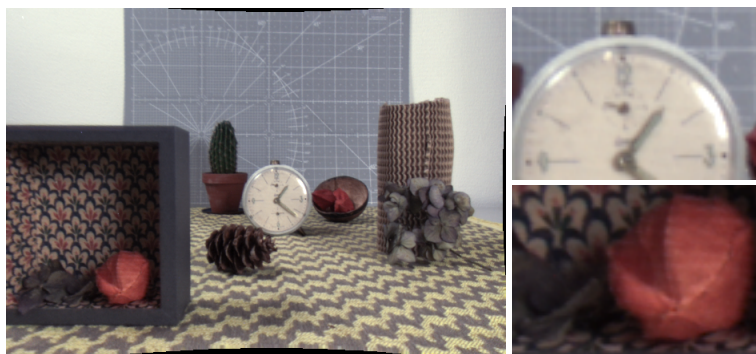
Figure 6.8 *minizoom* setting. Left: SAI from the center of the u, v -plane. Right: Details.



(a) Bok *et al.*



(b) Dansereau *et al.*



(c) Proposed generic light field reconstruction.

Figure 6.9 *minzoom* setting. Left: SAI from the edge of the u, v -plane. Right: Details.

plicable to all types of unfocused plenoptic cameras. Overall, this means that lens aberrations are not compensated for, which can be clearly seen in the barrel distortion in figure 6.8(b) and results in straight lines being bent. Yet again, when moving away from the center and looking at the SAIs at the edge of the u, v -plane, the quality of the images deteriorates, see figure 6.9. The reconstructed light field of Dansereau *et al.* and Bok *et al.* becomes much blurrier, while the quality of the generic method becomes only slightly worse. Strong vignetting artifacts appear in the upper left corner of the image, which (strangely) do not appear in the generic reconstruction, even though all methods are provided with the same devignetted sensor data. One possible explanation for this is that the vignetting increases the calibration error ε of the generic camera calibration. Rays with a high calibration error are superimposed by rays with a lower error, which leads to the compensation of the vignetting during the weighted interpolation of (6.13). Further, in the detail views, Dansereau *et al.* and Bok *et al.* show some pixels that are completely red, green, or blue, which are presumably dead pixels. These pixels do not appear in the generic reconstruction, since they also have a relatively high calibration error. So again, these pixels are efficiently suppressed by the weighted interpolation, and the missing information is obtained from neighboring rays.

6.3.1.3 Qualitative Evaluation of Epipolar Plane Images

Regardless of the quality of the reconstructed SAIs, the advantage of the proposed method becomes apparent in another area. Apart from the central view that only incorporates spatial information, the light field contains much more, *i.e.*, angular information. If one fixes an angular and a spatial coordinate in the 4D light field pointing in the same direction, *e.g.*, u and s , one gets a 2D slice of the light field, the so-called epipolar plane image (EPI), see Sec. 3.2. Lines of different slopes can be seen, whose orientation represents the depth of the observed object point. Depth estimation in light fields is thus reduced to a simple local orientation estimation in these EPIs, whereby the quality of the estimation is significantly influenced by the calibration. The higher the quality of the lines, the better the result of the depth estimation.

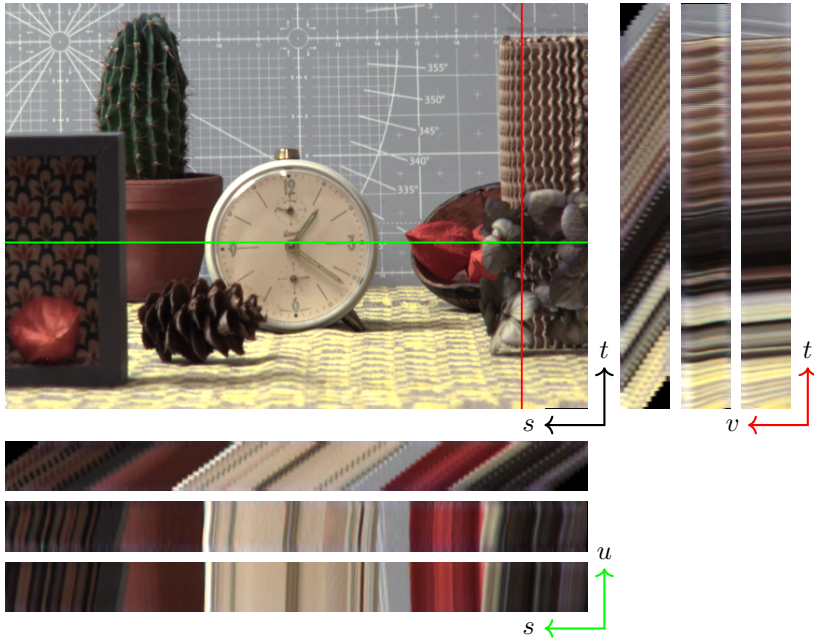


Figure 6.10 EPIs of the *maxzoom* setting in comparison: From top to bottom and left to right in the order Bok *et al.*, Dansereau *et al.*, proposed generic method.

For the *maxzoom* setting, figure 6.10 shows examples of horizontal and vertical EPIs generated by fixing u or v to its center coordinates and by selecting pixel lines for the s (red) or t (green) coordinate, respectively. The coordinates are chosen for each reconstructed light field to approximately be at the same position. The EPIs of Dansereau *et al.* show strong deviations from the ideal epipolar geometry, visible by the curvy epipolar lines. This is caused by the poor generalizability of the method which was developed for the old Lytro camera and works only moderately well for the newer Lytro Illum. Also, there are some errors at the top and the bottom. These areas correspond to pixels that are located at the boundary of the microlenses, where the imaging is more strongly distorted. For the EPIs reconstructed using the method of Bok *et al.* and the generic method, it can be seen that the epipolar geometry is reconstructed with higher

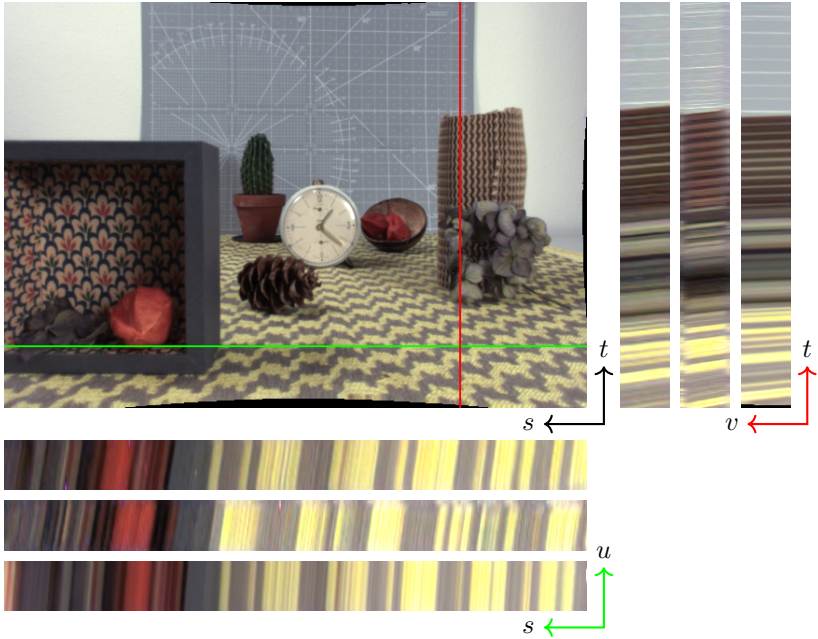


Figure 6.11 EPIs of the *minzoom* setting in comparison: From top to bottom and left to right in the order Bok *et al.*, Dansereau *et al.*, proposed generic method.

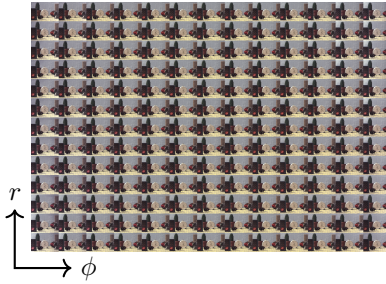
quality, observable by the straight lines. The slope of the epipolar lines is different for each reconstruction method, and it depends on the chosen parametrization of the light field. For the generic method, the general slope direction can be shifted by changing the distance f between the u, v - and s, t -plane. This does not change the information in the light field at all but only changes the “focal plane” of the light field, *cf.* Sec. 6.2.3. The parametrization of Bok *et al.* places the focal plane at infinity, hence the reconstructed light field can be interpreted as an array of virtual cameras with the optical centers of each camera being located at the same spatial pixel position. Thus, a point corresponding to zero slope results in zero disparity, which then theoretically implies a distance of infinity. As the EPIs show, the parametrization of the method of Dansereau *et al.* and the generic method seem to have the focal plane located near the alarm clock.

Example EPIs of the *minzoom* setting are depicted in figure 6.11. Now, the general slope direction is very similar and all methods show a good reconstruction of the epipolar geometry. Since the *minzoom* setting has less microlens vignetting, the reconstruction seems to work better for all methods. Only Dansereau *et al.*'s method still shows blurring in the upper and lower areas, and the reconstruction of Bok *et al.* shows distortions only in the most distant edge coordinates. However, while the epipolar geometry is reconstructed very well for all methods, only for the generic method and Bok *et al.*'s method, the distortions of the lenses are compensated, resulting in a rectified light field.

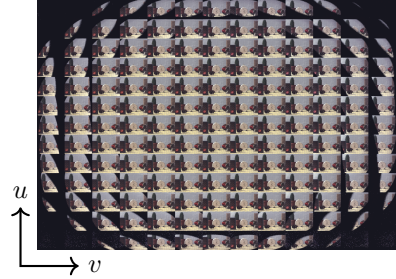
6.3.2 Comparison of Angular Sampling Grids

Another advantage of the proposed method is the free choice of sampling. Therefore, a more suitable sampling grid can be used. The polar sampling of the u, v -plane presented in Sec. 6.2.1 is better adapted to the data of the Lytro Illum light field camera, and can therefore better represent the light field. No unnecessary information is sampled and the result is more compact, or rather, more information is contained in the same amount of data. With the same resolution and thus the same size of the reconstructed light field, polar sampling effectively removes less information while representing the relevant information more accurately than Cartesian sampling. Figure 6.12 shows the comparison, whereby the light field is illustrated as an array of SAIs.

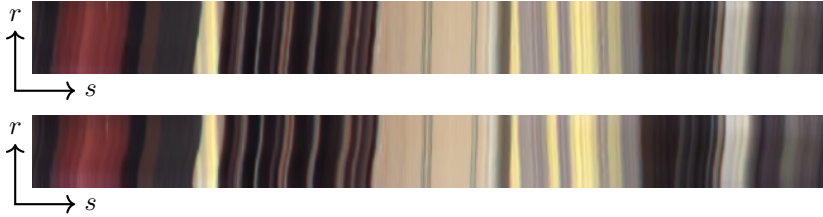
In detail, it is important how the polar sampling is implemented. As already described in Sec. 6.2.1, two options for the choice of radial sampling are considered. For the first choice, the radius is set in equidistant steps. For the second choice, the radius is set such that the pixel areas of all pixels of the sampling grid have equal size. This has the advantage that the signal-to-noise ratio remains the same for each pixel. Still, a minor disadvantage becomes apparent when analyzing the EPIs. Since now the step size of the radius is nonlinearly sampled, the lines in the EPIs are no longer straight but curved. The conventional light field depth estimation, which analyzes the slope of the lines, can therefore no longer be applied here without further consideration, as it would provide incorrect results or would make corresponding corrections necessary, *e.g.*, a local rescaling of the estimated slope of the lines. The comparison of the EPIs



(a) Polar sampling results in a more efficient representation of the data.



(b) Cartesian sampling reconstructs unnecessary peripheral areas of the u, v -plane.



(c) Top: Linear polar sampling with equidistant radius spacing. Bottom: Nonlinear polar sampling with equal pixel area.

Figure 6.12 Comparison of Cartesian and polar sampling. (a) and (b) The light field as an array of SAIs. (c) Polar EPIs for $\phi = 0$.

is shown in figure 6.12(c). In conclusion, it is therefore recommended to use polar sampling with equal pixel area if the light field camera is only used as a multi-view camera array. For use in the field of depth estimation, where the slope of the epipolar lines is analyzed, sampling the radius in equidistant steps is preferable.

6.3.3 Super-Resolution through Implicit Ray Interpolation

An interesting continuation of the generic light field reconstruction approach is the possibility to customize the dimension of the discrete pixel grid. This allows, for example, a light field super-resolution approach to be implemented in a very simple way. That is, the spatial resolution, the angular resolution, or both can be artificially increased. Of course, the

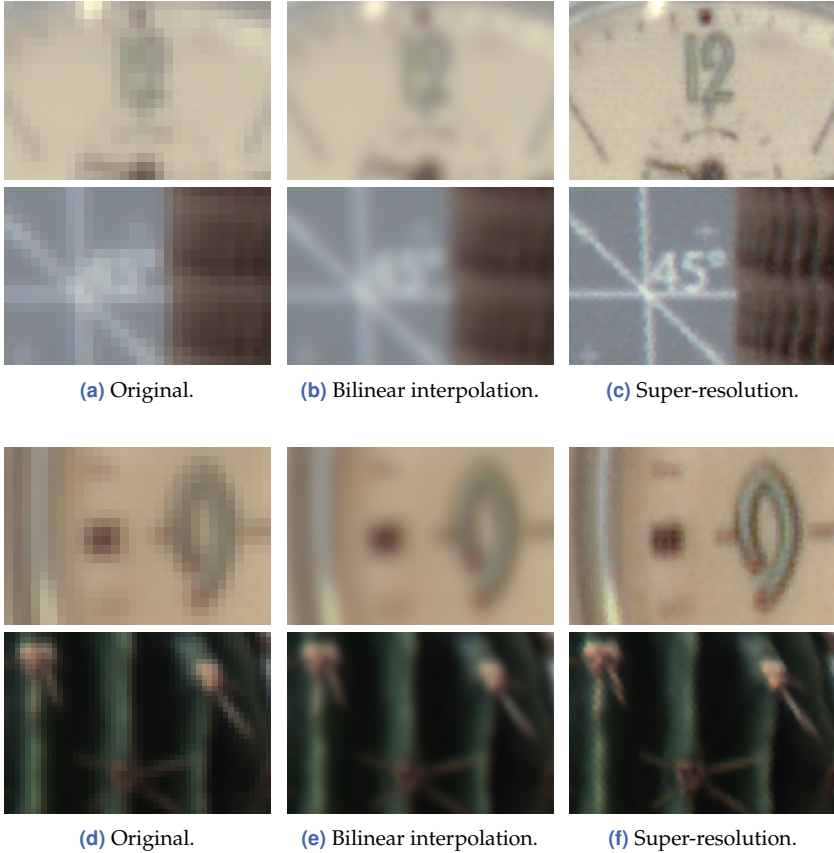


Figure 6.13 5×Super-resolution through implicit ray interpolation. (a)-(c) show details from the central SAI of the *minzoom* setting. (d)-(f) show details of the *maxzoom* setting. (a) and (d) show the original resolution of the light field. (b) and (e) show the result when bilinear interpolation is applied to the images. (c) and (f) show the result of the proposed generic super-resolution approach.

resolution cannot be indefinitely increased, since a corresponding ray is not always available for each super-resolved discrete coordinate. As the light field pixels become smaller with increasing resolution, fewer and fewer rays will hit a pixel. As a result, the reconstructed light field may contain holes. To fill these, the generic approach can now be used directly by considering neighboring rays. That means, in (6.12) $m > 1$ must be chosen.

An example super-resolved reconstruction of the light field using the *maxzoom* and *minzoom* settings is shown in figure 6.13. Here, $5\times$ super-resolution was applied spatially, resulting in the increased resolution of (15, 15, 3125, 2170). To interpolate missing data from the 4D neighborhood, $m = 4$ in the angular domain and $m = 7$ in the spatial domain are chosen. For comparison, a $5\times$ oversampling using bilinear interpolation on the SAIs is shown as well. While the bilinear interpolation increases the resolution, the result is still very blurry. The generic super-resolution approach, on the other hand, demonstrates impressively that the resolution could be considerably increased. Even small and distant details suddenly become clearly visible. The reason why the resolution of the images can be increased so much is due to the very high redundancy contained in light fields. Conventional super-resolution approaches must first estimate the depth of the scene and can subsequently map the scene points onto a virtual sensor [215, 218]. Alternatively, they are based on learning-based methods with complex CNN architectures [185, 235]. However, they all have in common that they require an already reconstructed light field. The advantage of the simple approach presented here is that none of this is necessary. Instead, super-resolved SAIs can be reconstructed directly from the generic ray bundle.

6.3.4 Light Field Reconstruction: Raytrix R5

While the generic method can already reconstruct light fields very well from the raw data of the Lytro Illum camera, it also works with other light field cameras without any further adaptation. To show this, the light field of a Raytrix R5 was reconstructed.

The Raytrix R5 light field camera has a monochromatic sensor of size 2048×2048 px with a pixel pitch of $5.5 \mu\text{m}$. In front of the sensor is an array of hexagonally arranged microlenses with about 25×25 px underneath

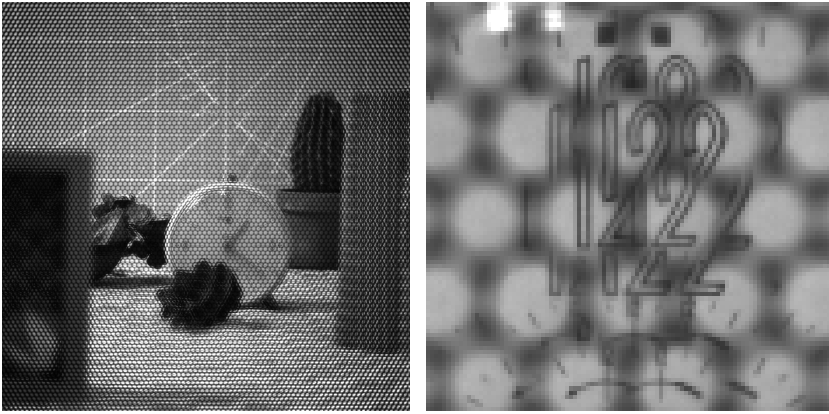


Figure 6.14 Raytrix R5: Raw sensor data and detailed view.

each microlens. A 35 mm fixed focal length objective with a hexagonally shaped aperture was used. The aperture is chosen such that the f-number matching is approximately fulfilled. The aperture cannot be rotated, and therefore it is not perfectly aligned with the hexagonal microlens grid, resulting in dark areas at the edge of the microlens, see figure 6.14. Because this camera is of the focused design, the distance between the microlenses and the sensor plane is different from the microlenses' focal length, see Sec. 3.2. In addition, the camera is a multi-focus plenoptic camera, which means that there are three types of microlenses, each with a different focal length.

As before, to transform the continuous light field parameters to a discrete pixel grid, the intersections of the camera rays with the u , v - and s , t -plane are analyzed. Figure 6.15 shows the histograms of the intersection points. The s , t -plane is quadratic due to the quadratic sensor, and the u , v -plane shows a circular distribution.

Because the Raytrix camera is a focused plenoptic camera, the number of pixels under each microlens no longer corresponds directly to the angular resolution. Rather, the microlenses now show micro-images of the scene. Each micro-image can therefore be interpreted as a virtual camera, where, depending on the position of the microlens, both the optical center of the micro-camera is shifted and a different small section

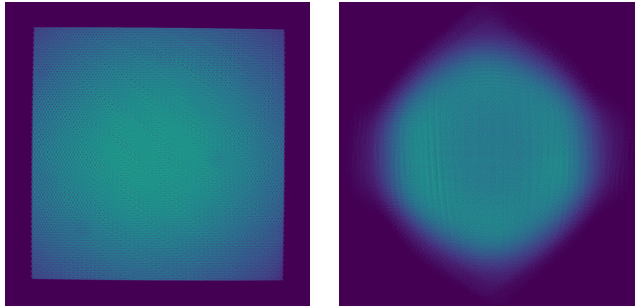


Figure 6.15 Raytrix R5: Histogram of ray-plane intersections. Left: s, t . Right: u, v .

of the scene is shown. The pixels below the microlens hence encode spatial information, while the microlens position contains both spatial and angular information. The angular resolution of the camera must therefore be roughly estimated. Because the micro-images in figure 6.14 show approximately a three-fold redundancy in both the horizontal and vertical direction, the angular resolution is chosen to be $N_u = N_v = 3$. The spatial resolution is slightly oversampled and set to $N_s = N_t = 1000$.

6.3.4.1 Qualitative Evaluation

The reconstruction of the central SAI is shown in figure 6.16. One can see that the scene is reconstructed correctly and that even details are recognizable. Since the Raytrix light field camera is built differently than the Lytro not everything in the reconstructed image is in focus. With this camera, the depth of field and the focus distance are now determined by the main lens and the main lens setting. Because the lens used in this experiment is not optimally selected for the Raytrix R5, strong vignetting effects are visible at the edges of the microlenses, as can be seen in the raw data, see figure 6.14. For the Lytro Illum camera, microlens-vignetting reduces the quality of the edge SAIs, whereas for the Raytrix the effect can theoretically also be seen everywhere in the central image. Very dark pixels at the edges of the microlenses cause reconstruction artifacts in the image due to a deconvolution operation. However, this unwanted effect could be resolved by using a more suitable lens with a hexagonal aperture, rotating the aperture to be aligned with the hexagonal grid, and

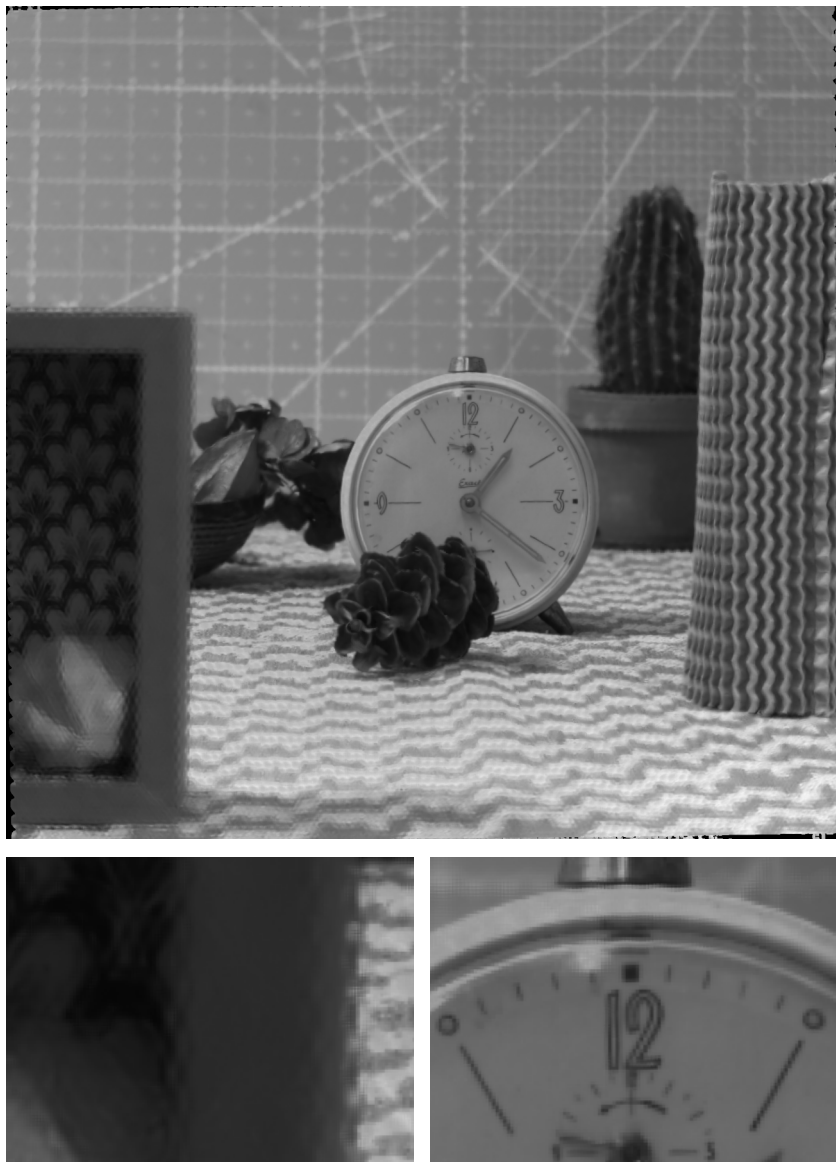


Figure 6.16 Raytrix R5. Top: SAI from the center of the u, v -plane. Bottom: Details.

manually adjusting the aperture's opening to the correct size. This effect is particularly strong in the lower-left area. While strong vignetting is visible here, an additional effect occurs. Because the camera is a focused plenoptic camera, the images under the microlens contain spatial information. The position of each microlens encodes both angular and spatial information. This has the consequence that the micro-images overlap to different degrees depending on the distance of the observed objects. *I.e.*, the degree of spatial redundancy seems to be distance-dependent. For the very close area at the bottom left, the micro-images do not overlap anymore and in addition, the strong vignetting creates perforated areas in which the scene cannot be observed completely. The missing information must therefore be interpolated from distant neighboring rays, which leads to the noticeable artifacts here.

A minor disadvantage of the generic light field reconstruction is that the multi-focus property of the Raytrix camera cannot be explicitly taken into account at first. This leads to blurred pixels being superimposed with sharp pixels in the reconstruction. Because the generic light field reconstruction in this work is intended to be completely independent of the observed scene and because it does not model the focal properties of the rays, this problem cannot be solved at first. However, one possibility to avoid this difficulty would be to classify the pixels beforehand and to assign them to the three categories of microlenses, *i.e.*, to the three focal lengths. With this, three separate light fields could be reconstructed for each microlens category, where of course each one would only observe a perforated part of the scene.

6.3.5 Light Field Reconstruction: K|Lens

Unlike the previous cameras, the K|Lens is not based on microlenses. To be more precise, the K|Lens light field camera is a light field objective lens that has to be mounted onto any full-sized camera sensor. For this experiment, the K|Lens was mounted on an Allied-Vision Prosilica GT4907C RGB sensor. The sensor has a resolution of 4864×3232 px with a pixel pitch of $7.4 \mu\text{m}$. Figure 6.17 shows the sensor image of the camera. The different views are clearly visible, which are mirrored differently by the kaleidoscope effect. Because the objective lens is not perfectly aligned with the sensor, the whole image array is slightly rotated.



Figure 6.17 K|Lens: Sensor image.

As with the other cameras, the K|Lens was calibrated using the generic calibration and then the intersections with the u, v - and s, t -planes were calculated. Figure 6.18 shows the histograms. The s, t -plane is rectangular, while the u, v -plane consists of 3×3 small dots. These dots correspond to the optical centers of the respective 3×3 views. The dots have a faint butterfly-shaped boundary, which is most likely caused by lens distortions having the consequence that there is no single center of projection. The choice of the discrete light field grid is very straightforward for the K|Lens. The angular dimension is chosen to be $N_u = N_v = 3$ and the spatial dimension is given as one third of the sensor resolution with $N_s = 1621, N_t = 1077$.

6.3.5.1 Qualitative Evaluation

For the light field reconstruction, only the direct neighbor was considered in the angular domain with $m = 1$, while second-order neighbors were considered in the spatial domain with $m = 2$. The light field reconstruc-

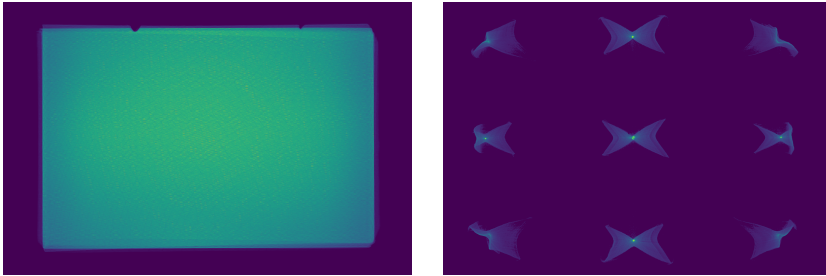


Figure 6.18 K|Lens: Histogram of ray plane intersections. Left: s - t -plane. Right: u , v -Plane. For better visualization, the colormap of the u , v -histogram has a logarithmic scale.

tion for the K|Lens camera as an array of SAIs is shown in figure 6.19. All in all, one can see that the different views of the camera are reconstructed very well. Since for the generic calibration, the arrangement of the pixels on the sensor is completely irrelevant, and since only the rays outside the camera are of importance, the kaleidoscope effect is automatically compensated by the generic reconstruction. In addition, due to the normalization of the generic ray bundle, the slight rotation of the K|Lens objective with respect to the sensor is corrected. Looking at the results in more detail, see figure 6.20, there are hardly any differences between the sensor data and the reconstruction, both in the central view and in the SAIs at the edge.

6.3.6 Camera Intrinsics and Calibration Error

Apart from the reconstruction of the light field and the qualitative analysis of the result, an exact characterization of the ray geometry is essential in many areas of computer vision, for optical metrology in general, as well as for deflectometry in particular. Since the presented method is based on generic camera calibration and to be comparable with the very same, the ray re-projection error ε from Sec. 5.5.1 needs to be investigated. This error corresponds to the distance between a geometric camera ray and an observed point on a reference target. To evaluate the error experimentally, a commercially available monitor was used as a reference target, whose pixels serve as reference coordinates. The monitor was captured from

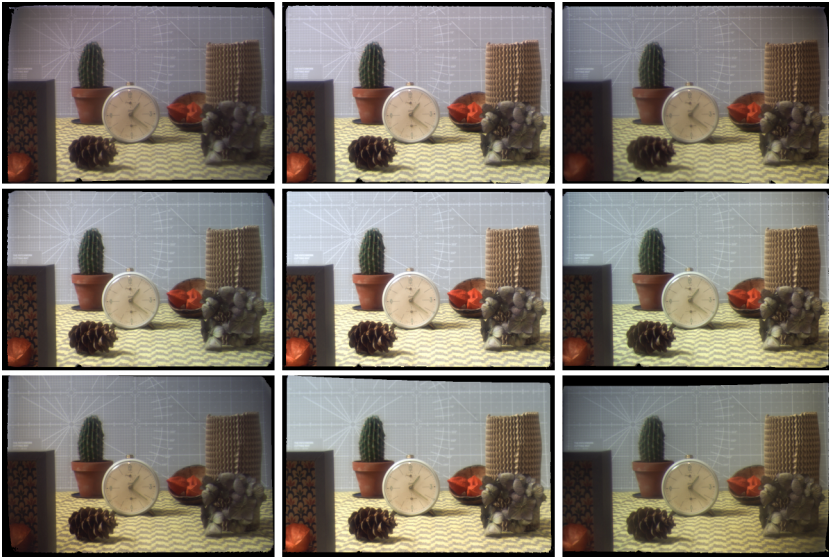


Figure 6.19 K|Lens: Reconstructed 3×3 light field.

different poses using the different cameras and camera configurations. In each pose, phase-shift features were acquired using the techniques from Ch. 4. For all cameras and both settings of the Lytro Illum camera, the raw data with the measured phase-shift features were converted to light fields using the presented generic reconstruction method. For comparison to the state of the art, the light fields corresponding to both Lytro camera settings were additionally reconstructed with the method by Bok *et al.* Further, with the help of the respective camera parameters, the camera rays could be determined for each light field. Subsequently, using these camera intrinsics and the generic pose estimation from Sec. 5.2.5, the 3D coordinates of the feature points were determined, and the ray re-projection error as an average value over all rays could be calculated.

The comparison of the different methods applied to the Lytro Illum is shown in table 6.1. The method of Dansereau *et al.* [44] could unfortunately not be evaluated, as the rectification algorithm and thus the determination of the camera parameters only works for the older Lytro but does not provide any meaningful results for the newer Lytro Illum.



(a) Center of sensor image.



(b) Center SAI.



(c) Upper right area of sensor image.



(d) Upper right SAI.

Figure 6.20 K|Lens: (a) and (c) Sensor image. (b) and (d) Generic reconstruction. For better visualization the mirroring of the sensor image is corrected in (c).

As expected, the generic calibration from Sec. 5.2 has the lowest calibration error, since each pixel can be calibrated individually and hence with high precision. However, this result cannot be compared directly to the other methods, since the correlations of the vision rays and the light field information are lost or cannot be used directly with this camera model. It is therefore only used to represent a lower limit of the calibration error. More importantly, the table shows that the presented generic light field reconstruction method has a much smaller mean error and RMSE than the method of Bok *et al.*, resulting in a better calibration with fewer outliers. And thus, the ray geometry is estimated much better although the qualitative comparison of the light field reconstruction for both methods is very similar. This is because the ray calibration of the presented generic light field reconstruction itself could be carried out very precisely, starting from the generic calibration. The nonidealities of the optics are accurately included in the generic camera model, and the

Table 6.1 Comparison of the ray re-projection errors for the Lytro Illum camera.

Lytro Illum	<i>minzoom</i> setting		<i>maxzoom</i> setting	
	ε in μm		ε in μm	
	Mean	RMSE	Mean	RMSE
Bok <i>et al.</i>	375.6	758.8	455.2	1117.7
Generic calibration	43.3	91.4	139.8	376.2
Generic LF-reconstruction	97.1	155.0	231.4	431.1

Table 6.2 Comparison of the ray re-projection errors for the Raytrix R5 and K|Lens.

Method	Raytrix R5		K Lens	
	ε in μm		ε in μm	
	Mean	RMSE	Mean	RMSE
Generic calibration	31.1	77.0	73.1	123.1
Generic LF-reconstruction	121.23	166.8	130.3	178.7

generic light field reconstruction only needs to sample the fully rectified light field from the resulting generic ray bundle. In contrast, the method by Bok *et al.* fits a low-dimensional camera model with a low-dimensional distortion model to the camera data. Deviations from this model cannot be taken into account, and therefore the calibration error increases. Even though the generic reconstruction is based on the generic calibration, the ray re-projection error is slightly worsened by the interpolation and rounding operations of Sec. 6.2.2. A direct comparison of both camera settings reveals that the calibration for the *maxzoom* setting provides slightly inferior results, regardless of the method. Due to the stronger microlens vignetting for this setting, as shown in figure 6.4, the peripheral areas of the microlenses capture much less light, which increases the uncertainty of the calibration features, and thus worsens the calibration.

Because the software for the Raytrix and the K|Lens are not available as open-source, only the result of the presented generic methods is shown here. Table 6.2 shows the results of the respective calibrations. Similar to before, the generic calibration can be seen here as a lower limit. For the K|Lens, it can be seen that during the light field reconstruction the

Table 6.3 Intrinsic camera parameters for the Lytro Illum’s *maxzoom* setting.

Bok <i>et al.</i> [20]	Generic LF-reconstruction
$f_s = 4295.7$	$f_s = 4295.3$
$f_t = 4295.7$	$f_t = 4295.3$
$c_s = 278.1$	$c_s = 273.9$
$c_t = 192.0$	$c_t = 194.3$
baseline = 2.99 mm	baseline = 3.16 mm

calibration error again worsens only by a small factor. In contrast to this, the generic calibration of the Raytrix R5 camera is very accurate. However, when reconstructing the light field from this, the error increases strongly. This is mainly due to the strong interpolation artifacts that can also be observed in the reconstructed SAIs, see figure 6.16. Nevertheless, the quality of the light field reconstruction of all cameras and all zoom settings is still very close.

For a detailed comparison between the presented method and the method of Bok *et al.*, the *maxzoom* light field was reconstructed at the same resolution as Bok *et al.*’s reconstruction. Since the central SAI of Bok *et al.* and the generic method are now very similar, their intrinsic parameters should also be comparable, given the same parametrization of the light field grid. Table 6.3 shows the camera parameters for the central image, as well as the baseline between neighboring SAIs. Both times the parameters are very similar, and the optical center (c_s, c_t) is estimated to be close to the center of the respective images. Hence, the proposed generic light field reconstruction yields reasonable results. The distance between the SAIs is similar too, with a slightly larger baseline for the generic reconstruction. For Bok *et al.*, this results in a camera array of width 35.88 mm, while for the generic reconstruction it results in a width of 37.92 mm. This again means that Bok *et al.* does not capture the outermost regions of the main lens, while the presented generic method captures a slightly larger area in the parametrization investigated here.

Even if the parameters of the two methods are very similar, this does not mean that the quality of the calibration must be comparable, since the reconstruction of the light field is different. By taking a closer look at the reconstruction quality of the Lytro Illum reconstruction, it can be

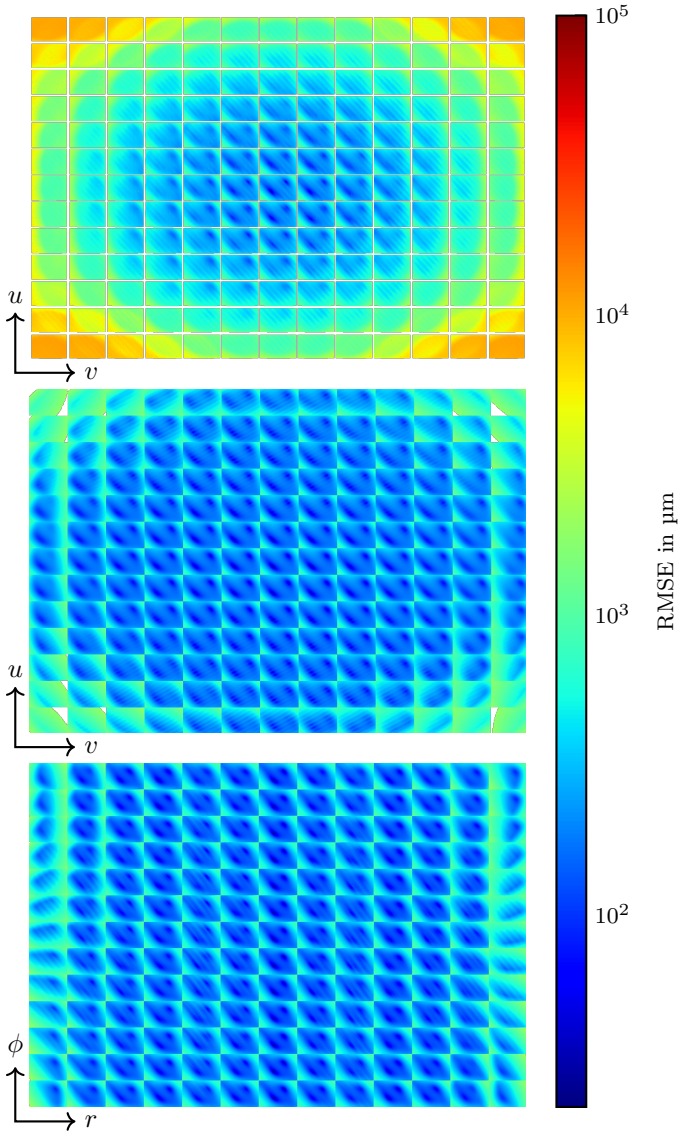


Figure 6.21 Lytro Illum maxzoom setting: Ray re-projection error per pixel for all SAIs. Top: LF-reconstruction by Bok *et al.* [20]. Middle: Generic LF-reconstruction with Cartesian angular sampling. Bottom: Generic LF-reconstruction with polar angular sampling.

seen that the errors increase the further the SAIs are from the center. This effect is particularly strong with the calibration of Bok *et al.*, while it is much smaller with the generic reconstruction. Figure 6.21 shows the comparison. Bok *et al.*'s reconstruction shows for the central SAI a very high calibration quality with small ray re-projection errors. However, the quality decreases strongly towards the outer regions. Only about the inner 9×9 SAIs still have an RMSE value smaller than $300 \mu\text{m}$. The generic reconstruction, on the other hand, shows small errors even up to the outer regions of the u, v -plane. Only at the very outer limits does the error increase. At the same time, invalid pixels appear. Due to the Cartesian sampling of the u, v -plane, areas outside the main lens are now also sampled where simply no rays exist. However, these pixels do not necessarily pose a problem, since they can simply be classified as invalid. To prevent such issues from occurring altogether, one can simply use a polar parametrization of the angular coordinates. Thereby only the relevant areas of the main lens are sampled and invalid pixels are avoided while maintaining a comparable calibration quality. Here again, only at the most distant radius values, the error slightly increases.

6.4 Summary

This chapter presented a method to calibrate any light field camera (*e.g.*, microlens-based, mirror-based, camera arrays) without having to model any optical properties explicitly. Utilizing a generic calibration, the individual camera rays were precisely calibrated. Since conventional light field-related algorithms require regular sampling, the method transformed the result into an equivalent light field representation and fitted a regular 4D grid onto the irregular camera rays. The summation of the weighted intensity values of the rays finally led to the interpolation and reconstruction of a rectified light field. Apart from the usual Cartesian sampling of the angular coordinates, this chapter presented two possibilities to sample them in polar coordinates. This proved to be advantageous since the light field information can now be represented more compactly. Besides the pure reconstruction of the light field's radiometric quantities, a derivation of the intrinsic camera parameters was also presented, *i.e.*,

the geometric quantities. The reconstructed light field can therefore easily be used in any subsequent application.

Eventually, experiments showed that the proposed method can provide good reconstructions and rectified light fields. The epipolar geometry between the sub-aperture images is preserved and even shows better results than the conventional state-of-the-art methods. In addition, an analysis of the geometric parameters utilizing the ray re-projection error showed that the proposed method has a smaller calibration error than the state-of-the-art methods from the literature, and thus, it achieves a better calibration. While providing very good results for a classical unfocused plenoptic camera, the evaluation demonstrated that the generic reconstruction works for many kinds of light field cameras and yields a highly accurate calibration. For the K|Lens, the generic light field reconstruction is perhaps not the best solution, since the camera optics are in principle not very complex and the generic camera calibration is quite time-consuming due to the necessary acquisition of dense features. Therefore, simpler models with conventional distortion models would perhaps find a similarly satisfying solution for this specific camera. However, the results clearly show that the presented generic light field reconstruction achieves very high accuracy for any light field camera system, no matter if it is microlens-based, mirror-based, or relies on other techniques.

In summary, both the information of the observed scene and the geometric structure of the light field are preserved by adequate rectification and calibration. And in the end, a better reconstruction of the light field and an improved estimation of the camera's geometrical properties leads to better results when used in optical metrology or depth estimation.

7 Specular Surface Reconstruction

The deflectometric registration already enables a visual inspection of specular objects with the possibility to detect local surface defects or to roughly classify shape deviations. However, it is not yet sufficient to enable a deflectometric 3D reconstruction. If, in addition, the intrinsic calibration of the camera and the monitor as well as the extrinsic calibration of the measurement setup are known, a normal field can in principle be determined from the deflectometric measurements. However, the three-dimensional shape of a specular object cannot be directly determined for the time being, even if a calibrated setup is used. As shown in Sec. 3.1, a possible surface normal can be calculated for each point in the camera's field of view, so an infinite number of possible surfaces could be the cause of the same measurement. Because of this ambiguity, it is necessary to use regularization methods that can determine the true surface normal and thus lead to an unambiguous solution. The reconstruction of the specular surface is usually done in two steps. In the first step, the ambiguity of a single deflectometric measurement is resolved by considering additional data. The result is an approximate position of the surface in terms of points in space and the corresponding normal vectors of the surface at these points. Even if a solution for the surface is already available through this regularization, its accuracy is typically still insufficient for practical applications. Because deflectometry is a slope measuring technique, the accuracy of the normal estimate is magnitudes higher than the measurement of the depth. The actual specular surface reconstruction is therefore performed as a secondary step. Here, the low accuracy surface points obtained from the regularization and the corresponding high accuracy normal vectors are taken and combined to produce a smooth and continuous representation of the surface.

Since this work deals with light field cameras, Sec. 7.1 describes procedures that use the properties of these cameras to enable a regularization of the deflectometric ambiguity. Subsequently, Sec. 7.2 presents an

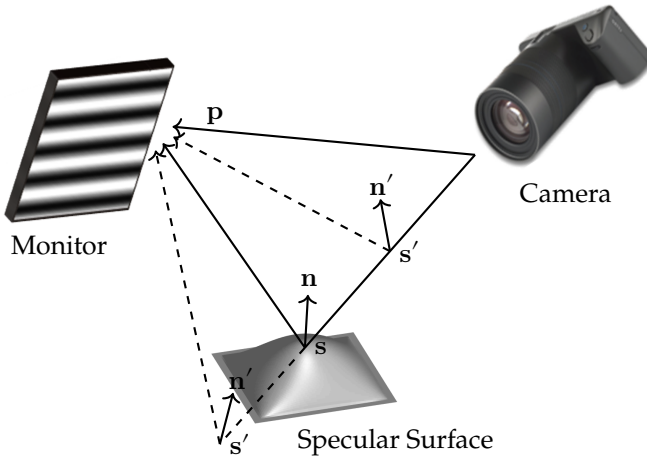


Figure 7.1 Ambiguity of the deflectometric normal estimation. Even with a fully calibrated system and by knowing the coordinate of the observed reference feature, a potentially valid surface normal can be calculated for every point on a camera ray.

algorithm that fuses the regularization data with the normal estimates to obtain a high accuracy surface reconstruction. Finally, Sec. 7.3 evaluates the presented methods by using an experimental deflectometry setup to reconstruct the shape of different specular objects.

7.1 Deflectometric Regularization

As explained before, the deflectometric reconstruction of the normal field is ambiguous. Therefore, initially, no unique solution for the specular surface can be specified, see figure 7.1. To resolve the ambiguity of the deflectometric measurement, additional regularizing information is needed. In principle, it is sufficient to measure only the distance to one point of the surface and to reconstruct the surface from the normal field starting from this point [11]. Though, if more measurements are available, this can help to reduce the influence of a single uncertain and noisy surface point. For this purpose, various procedures were introduced in Sec. 3.1.2, all of which require a more or less complex system structure.

The main focus of this thesis is to efficiently use the special properties of the light field camera to enable a deflectometric reconstruction of the surface. In this section, two methods are presented in which the light field camera can be directly used to obtain additional information about the surface, which has a regularizing effect on the deflectometric measurement.

7.1.1 Light Field Depth-Based Regularization

Since the light field camera can partially capture the light field of the observed scene, it can extract much more information than a standard camera. The additional information, in contrast to standard cameras, allows changing the perspective on the scene after the exposure, thus enabling depth information to be extracted.

The depth of a diffusely reflecting scene, *i.e.*, the distance of an observed object point, can be determined by analyzing the light field's geometric structure, *i.e.*, the slope of the epipolar lines in the EPIs (*cf.* Sec. 3.2). The light field camera can therefore be used as a compact passive 3D camera, meaning that structured illumination is not required. When surveying partially reflecting surfaces, the special properties of the light field camera allow finding depth features on the direct surface as well as determining the depth of the reflected scene. These independent measurements can be used as an additional source of regularizing information for deflectometry.

In the following, it is demonstrated how the depth estimation of the light field camera can be used to solve the ambiguity problem of the deflectometric normal reconstruction.

7.1.1.1 Direct and Indirect Depth Estimation

The depth estimation of light field cameras allows to find candidates for possible surface points, and it thus makes it possible to resolve the ambiguity of the deflectometric normal estimation. In practice, two situations arise, see figure 7.2. First, if the surface of the measurement specimen has diffusely reflecting regions, a standard light field-based depth estimation can be used to directly measure the distance between the camera and the surface for each pixel (or camera ray), see Sec. 3.2. The set of

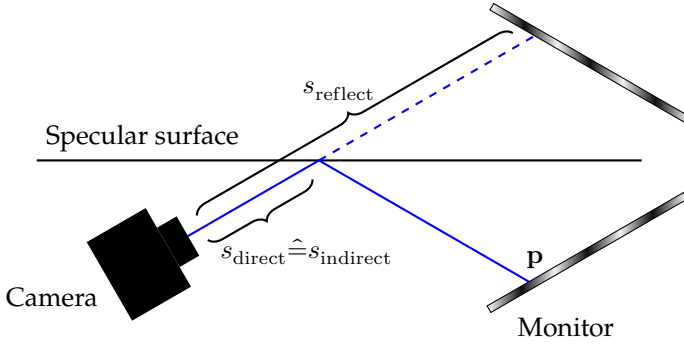


Figure 7.2 Direct depth estimation detects diffuse features on the surface. Indirect depth estimation estimates the depth to the reference monitor and calculates the depth to the surface using the known measurement setup.

depth values z_{direct} thereby estimates the distance to the surface. With the intrinsic parameters of the light field camera and the forward projection model (6.27), this depth can be transformed to a corresponding ray length, which can then be directly used to regularize the deflectometric normal estimate:

$$s_{\text{direct}} \hat{=} \|s(z_{\text{direct}})\| . \quad (7.1)$$

If the measurement sample is fully specular, the light field camera is not able to directly determine the distance to the surface. The real surface is virtually invisible. For plane mirrors, the camera will instead estimate the distance to the reflected reference scene. The resulting ray length becomes

$$s_{\text{reflect}} = \|s\| + \|s_r\| . \quad (7.2)$$

Nevertheless, with the help of the knowledge about the calibrated deflectometric measurement setup and with the registration of camera rays to monitor pixels, the direct distance to the surface can be calculated from the indirect depth measurement. It follows with the deflectometric measurement $\mathbf{p} = \mathbf{s} + \mathbf{s}_r$ and the depth estimate $s_{\text{reflect}} = \|s\| + \|s_r\|$:

$$\|s_r\|^2 = \|\mathbf{p} - \mathbf{s}\|^2 = \|\mathbf{p}\|^2 - 2\mathbf{p}^T \mathbf{s} + \|s\|^2 , \quad (7.3)$$

$$\|s_r\|^2 = (s_{\text{reflect}} - \|s\|)^2 = s_{\text{reflect}}^2 - 2s_{\text{reflect}} \|s\| + \|s\|^2 . \quad (7.4)$$

Equating (7.3) and (7.4), and using $\hat{\mathbf{s}} = \frac{\mathbf{s}}{\|\mathbf{s}\|}$ gives an estimate s_{indirect} for the distance to the surface:

$$s_{\text{indirect}} \hat{=} \|\mathbf{s}\| = \frac{1}{2} \frac{\|\mathbf{p}\|^2 - s_{\text{reflect}}^2}{\mathbf{p}^T \hat{\mathbf{s}} - s_{\text{reflect}}} . \quad (7.5)$$

However, it must be mentioned that (7.2) is valid only if the mirror surface is sufficiently flat or if the observation angle is chosen appropriately. Investigations of Criminisi *et al.* [41] and Swaminathan *et al.* [195] have shown that the measured length appears compressed or stretched in contrast to the true length depending on the surface shape and the measurement configuration. That is, in reality, the estimated depth becomes

$$s_{\text{reflect}} = \|\mathbf{s}\| + \alpha^{-1} \|\mathbf{s}_r\| , \text{ with } \alpha = 1 + 2 \|\mathbf{s}_r\| \kappa \cos(\beta) , \quad (7.6)$$

where the multiplicative factor α in the depth estimate is affected by the distance of the reference scene $\|\mathbf{s}_r\|$, the incidence angle β between camera ray and surface normal, and the curvature of the surface κ . Here, the curvature is measured relative to the “direction of motion” of the camera, which corresponds in a light field camera to the direction of the used EPI. That is, different EPIs may provide different depth estimates. Since it is not possible to estimate the values for κ and β without further knowledge about the surface and the measurement setup, the only solution to this issue is to detect regions of strong curvature and to exclude them from being used for regularization. Even though the surface cannot be reconstructed unambiguously in deflectometry without prior regularizing data, indications about the curvature of the surface can still be obtained directly from the deflectometric measurement. With an increase in local surface curvature, the directional derivatives of the registration data increase as well [100]. Hence, a simple second-order gradient calculation with subsequent thresholding allows the detection of high curvature regions.

The two-fold depth estimation presented in this section can in principle be performed at the same time due to the special properties of the light field camera. When light field cameras observe partially reflecting or transparent objects, the resulting light field can be interpreted as a superposition of two individual light fields. For classical stereo camera

systems, this is usually troublesome and results in erroneous depth estimates. For light fields, however, an analysis of the EPIs now shows a superposition of the line-like structures as well [96]. A simultaneous estimate of both orientations thus provides a depth estimate of both the partially specular object and the reflected scene. Methods for estimating these depths use, *e.g.*, higher-order structure tensors or optical flow [217, 232]. In practice, however, it became apparent that this simultaneous depth estimation is not suitable for deflectometric regularization and that a sequential estimation leads to better results since the task is simplified. When examining surfaces with diffuse components, it is only necessary to take an image where the monitor is completely white, and thus, the surface is sufficiently well illuminated. Since the reflection (the monitor) now contains no structure, the depth estimation algorithm will only detect features directly on the surface. To subsequently measure the distance to the reflected monitor, it is possible to perform a depth estimation directly on the registration data. This means that in this case the light field does not contain color information, but each light field pixel is assigned the 2D coordinates of the observed monitor pixel estimated via phase-shift coding. Using this as a direct image feature is advisable because then image noise is drastically reduced, enabling a more robust depth estimation.

In summary, for partially specular surfaces, the light field camera can obtain two separate depth estimates. However, most of the classical depth estimation algorithms (including the ones based on CNNs) only provide the depth of the central SAI [98, 187], since it yields the most accurate results. Further, many algorithms provide an additional confidence estimate for the depth [18, 199]. Hence, for any partially specular surface, the direct depth estimate z_{direct} with confidence c_{direct} is obtained. Areas with high confidence are caused by a structured surface, while low confidence implies areas with little structure or even fully specular areas. For planar mirrors, the indirect depth estimate z_{indirect} can be obtained with confidence c_{indirect} . In contrast to the direct depth estimation, the confidence is hereby lower for diffusely structured surface areas, while fully specular areas have higher confidence. For non-planar mirrors, the confidence measure is also affected by the curvature.

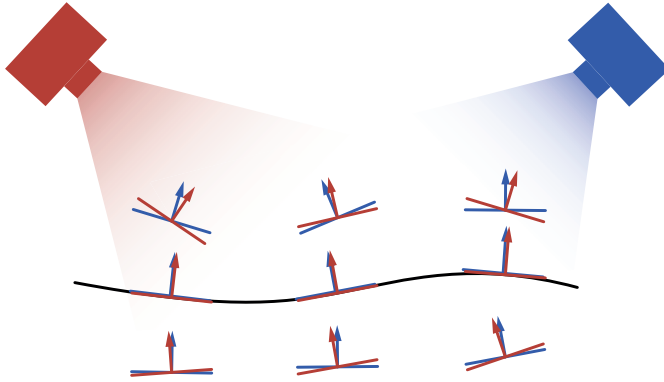


Figure 7.3 Principle of stereo deflectometry: A deflectometric measurement induces two independent normal fields in the fields of view of the cameras. On the true surface, the surface normals measured in both cameras must coincide.

7.1.2 Light Field Multi-View-Based Regularization

A shortcoming of the regularization method from the previous section is that it only estimates the depth of the central SAI and does not provide the depth for the other SAIs. However, the major disadvantage of depth-based regularization is that it only works for special surfaces. This means that initially it cannot be used to measure fully specular and curved surfaces. To be able to measure such surfaces as well, this section introduces a combination of the principle of multi-stereo deflectometry with light field cameras to obtain accurate regularization points in each SAI.

In (multi-)stereo deflectometry, the surface is observed by at least one additional camera. In contrast to the classical stereo vision and the depth estimation of diffuse surfaces, on fully specular surfaces there is the difficulty that no direct point correspondences can be found since initially only virtual features are captured in both cameras. That is, pixels from the cameras observing the same surface point will see different points in the monitor plane. However, specular stereo can be achieved by correlating the normal vector fields induced by two measurements, where the true surface can be found in the intersection of both solution manifolds. Hence, an indirect surface triangulation can be achieved with the following: In the field of view of the first camera a three-dimensional

normal field \mathbf{n}_1 is induced by a deflectometric measurement. The second camera with a different field of view on the test object provides another normal field \mathbf{n}_2 . Thus, for each point in the intersection of the fields of view, two candidates for surface normals can be calculated. On the real test surface, these normals must coincide $\mathbf{n}_1 = \mathbf{n}_2$. For points that are not on the surface, one usually observes a deviation of the normal directions [10]. Figure 7.3 illustrates this principle.

A very basic algorithm for surface reconstruction is to determine along a search direction the points where the two normal directions coincide best. These points regularize the deflectometric ambiguity and represent possible surface points. The normals determined in this way are the corresponding surface normals. The stereo principle can be easily extended to a multi-view approach. And since the light field camera can be interpreted as a multi-camera array, a light field multi-view-based regularization can be easily implemented, where surface points can be found for each SAI.

7.1.2.1 Regularization by Normal Disparity Minimization

To be able to quantitatively evaluate the similarity of the measured surface normals for each point in space, a suitable distance measure, the so-called normal disparity, has to be defined. A disparity measure that is widely used in the literature is the variance of the normal field in the observed surface point under consideration [10, 21]. This can be obtained by first calculating the average of the normal estimates corresponding to every view $\mathbf{n}_{\text{mean}} = \frac{1}{N} \sum_{n=1}^N \hat{\mathbf{n}}_n$, and by subsequently calculating the mean angle between this mean normal and the individual normals:

$$J(\mathbf{s}) = \frac{1}{N} \sum_{n=1}^N \arccos \left(\hat{\mathbf{n}}_n^T(\mathbf{s}) \hat{\mathbf{n}}_{\text{mean}}(\mathbf{s}) \right)^2, \quad (7.7)$$

where $\hat{\mathbf{n}} = \frac{\mathbf{n}}{\|\mathbf{n}\|}$ indicates a unit vector, and where all normal estimates obviously depend on the examined point \mathbf{s} .

If additional information about the quality of the individual measurements is available, it is reasonable to use weighted averages instead of plain averages. The quality of the estimation is influenced by two factors: the uncertainty of the deflectometric registration, *i.e.*, the phase uncertainty σ , and the inherent accuracy of the camera calibration, *i.e.*, the

residual calibration error ε . Thus, a weighting factor combining both factors can be provided for every camera pixel, or in the case of the light field, it is available for every ray $\hat{\mathbf{s}}_{uv}(s, t)$ of each SAI:

$$w_{uv}(s, t) = \frac{1}{\sigma_{\varphi}^2(u, v, s, t) \varepsilon^2(u, v, s, t)} . \quad (7.8)$$

For the sake of brevity, the dependence on the individual light field pixels is omitted in the following, as long as it does not impede understanding. Hence, by interpreting the light field as a camera array, for every spatial pixel (s, t) , the objective that needs to be minimized becomes

$$J(\mathbf{s}_{uv}) = \frac{1}{\sum_{u,v} w_{uv}} \sum_{u,v} w_{uv} \arccos \left(\frac{\hat{\mathbf{n}}_{uv}^T(\mathbf{s}_{uv}) \sum_{u,v} w_{uv} \hat{\mathbf{n}}_{uv}(\mathbf{s}_{uv})}{\left\| \sum_{u,v} w_{uv} \hat{\mathbf{n}}_{uv}(\mathbf{s}_{uv}) \right\|} \right)^2 . \quad (7.9)$$

To find the surface, it is now necessary to search the entire measurement space for the regions with minimum normal disparity. To avoid discretizing the measurement space with unnecessarily high resolution, and to prevent a too coarse representation of the surface as well, initially, no continuous parametrization of the surface is sought. Instead, the exact resolution of the camera is used, and the optimal distance to the surface is searched for each camera pixel, *i.e.*, for each ray. As a consequence, the minimization of the normal disparity along each ray depends on only one parameter: the length of the ray or rather the depth z of the corresponding point $\mathbf{s}(z)$. Moreover, each ray can be considered individually, which allows the optimization to be performed in parallel. For each pixel, respectively for each camera ray, one obtains the one-parametric optimization problem

$$z = \arg \min_z J(\mathbf{s}(z)) . \quad (7.10)$$

To evaluate $J(\mathbf{s})$ and to calculate the disparity, a few intermediate steps are necessary. First, starting with a single discrete light field pixel (u, v, s, t) , the corresponding point in space $\mathbf{s}_{uv}(z)$ must be determined according to the current evaluated depth z . For this purpose, the spatial

pixel (s, t) of the current SAI is lifted into space by using the camera intrinsics from Sec. 6.2.3:

$$\mathbf{s}_{uv}(z) = z \cdot \mathbf{K}_{uv}^{-1} \begin{pmatrix} s \\ t \\ 1 \end{pmatrix} - \mathbf{t}_{uv}. \quad (7.11)$$

Subsequently, with the help of (6.26), the same point is then projected back onto the virtual sensor planes of all other SAIs with the angular coordinates (\bar{u}, \bar{v}) :

$$\begin{pmatrix} s_{\bar{u}\bar{v}} \\ t_{\bar{u}\bar{v}} \\ 1 \end{pmatrix} = \frac{1}{z} \mathbf{K}_{\bar{u}\bar{v}} (\mathbf{s}_{uv}(z) + \mathbf{t}_{\bar{u}\bar{v}}). \quad (7.12)$$

With the help of the projected light field pixel coordinates, the respective deflectometric measurement can be obtained consisting of the measured monitor coordinate that is transformed to the camera coordinate system and the value of the respective weighting factor as well:

$$\mathbf{p}_{\bar{u}\bar{v}} = \mathbf{R}\mathbf{x}(\bar{u}, \bar{v}, s_{\bar{u}\bar{v}}, t_{\bar{u}\bar{v}}) + \mathbf{t}, \quad (7.13)$$

$$w_{\bar{u}\bar{v}} = w(\bar{u}, \bar{v}, s_{\bar{u}\bar{v}}, t_{\bar{u}\bar{v}}). \quad (7.14)$$

Due to the possibility of non-integer spatial pixels $(s_{\bar{u}\bar{v}}, t_{\bar{u}\bar{v}})$, intermediate values are calculated by means of bilinear interpolation. In the final step, the surface normals are calculated using the surface point under consideration $\mathbf{s}_{uv}(z)$, the through phase-shift coding measured monitor points $\mathbf{p}_{\bar{u}\bar{v}}$ and the respective camera rays $\hat{\mathbf{s}}_{\bar{u}\bar{v}}$ for all SAIs (including u, v):

$$\mathbf{n}_{\bar{u}\bar{v}}(z) = \frac{\mathbf{p}_{\bar{u}\bar{v}} - \mathbf{s}_{uv}(z)}{\|\mathbf{p}_{\bar{u}\bar{v}} - \mathbf{s}_{uv}(z)\|} - \hat{\mathbf{s}}_{\bar{u}\bar{v}}. \quad (7.15)$$

Using these steps, the normal disparity (7.9) can be calculated for the pixel (u, v, s, t) at the depth z .

The one-parametric optimization problem (7.10) can now be optimized along the individual camera rays using a line search algorithm. Since the computation of the normal disparity is costly, gradient-free methods such as *Brent's method* are suitable for this purpose [30]. This method combines golden-section-search with parabola approximations, and it

converges in the ideal case with a quadratic rate to the optimum. In each evaluation step of the optimization, the normal disparity (7.9) must be calculated for the current depth value z . And due to the independence of the individual camera pixels, the corresponding depth values can be easily optimized in parallel. However, since a fully convex objective is required for a correct optimization, a few issues arise regarding the disparity minimization. In general, the depth-dependent normal disparity has at least two minima. One appears at the surface. Another one emerges for $z \rightarrow \infty$, which is due to camera rays being gradually more parallel to each other for greater distances and thus the surface normals being calculated to become more equal. An incorrect initialization could therefore lead to an erroneous depth estimate [200]. For particular imaging configurations and concave surfaces, the issue becomes even worse, since then the objective may even show multiple minima. More precisely, different surfaces can be generated which cannot be distinguished even with a stereo approach [221]. To solve these difficulties, prior knowledge about the distance to the surface must be used, and the minimization must be constrained by boundary conditions. Consequently, the final optimization problem is obtained for each pixel, respectively for each ray, where the search space of the depth z is constraint by a convenient choice of bounds $[z_{\min}, z_{\max}]$ that avoids incorrect minima:

$$z = \arg \min_{z \in [z_{\min}, z_{\max}]} J(s(z)) . \quad (7.16)$$

Hence, with the same notation as for the other regularization points, the depth map z_{multi} is obtained and can be used to regularize the deflectometric normal measurement. Alg. 3 summarizes the multi-view disparity minimization.

7.2 Surface Reconstruction

In principle, the regularization points which can be found with the methods from the previous sections can be used directly to reconstruct the surface, for example, by calculating an average. However, since multi-stereo measurement systems such as the light field camera are limited in their measurement quality by the width of the effective stereo baseline,

Algorithm 3 Light Field Multi-Stereo Deflectometry**Input:** Registration data, camera intrinsics, relative pose**Output:** Depth and surface normal with minimal normal disparity**Initialize:** Set min and max distance

```

1: for  $(u, v, s, t) \in [0, N_s - 1] \times [0, N_t - 1] \times [0, N_u - 1] \times [0, N_v - 1]$  do
2:   Get first depth value (using Brent's method)
3:    $z := z_{uv}(s, t) \leftarrow \text{Brent}(z_{\min}, z_{\max})$ 
4:   while Disparity  $J$  is not yet sufficiently small do
5:     Project ray to world coordinates with for depth

6:      $\mathbf{s}_{uv}(z) = z \cdot \mathbf{K}_{uv}^{-1} \begin{pmatrix} s \\ t \\ 1 \end{pmatrix} - \mathbf{t}_{uv}$ 

7:     Calculate disparity and surface normal
8:     for  $(\tilde{u}, \tilde{v}) \in [0, N_u - 1] \times [0, N_v - 1]$  do
9:       Transform to SAI-pixel coordinates
10:       $\begin{pmatrix} s_{\tilde{u}\tilde{v}} \\ t_{\tilde{u}\tilde{v}} \\ 1 \end{pmatrix} = \frac{1}{z} \mathbf{K}_{\tilde{u}\tilde{v}} (\mathbf{s}_{uv}(z) + \mathbf{t}_{\tilde{u}\tilde{v}})$ 

11:      Get corresponding monitor coordinate and weight factor
12:       $\mathbf{p}_{\tilde{u}\tilde{v}} = \mathbf{R}\mathbf{x}(\tilde{u}, \tilde{v}, s_{\tilde{u}\tilde{v}}, t_{\tilde{u}\tilde{v}}) + \mathbf{t}$ 
13:       $w_{\tilde{u}\tilde{v}} = w(\tilde{u}, \tilde{v}, s_{\tilde{u}\tilde{v}}, t_{\tilde{u}\tilde{v}})$ 
14:      Calculate surface normal
15:       $\mathbf{n}_{\tilde{u}\tilde{v}}(z) = \frac{\mathbf{p}_{\tilde{u}\tilde{v}} - \mathbf{s}_{uv}(z)}{\|\mathbf{p}_{\tilde{u}\tilde{v}} - \mathbf{s}_{uv}(z)\|} - \hat{\mathbf{s}}_{\tilde{u}\tilde{v}}$ 
16:    end for
17:     $J(\mathbf{s}_{uv}(z)) = \frac{1}{\sum_{u,v} w_{uv}} \sum_{u,v} w_{uv} \arccos \left( \hat{\mathbf{n}}_{uv}^T(z) \frac{\sum_{u,v} \hat{\mathbf{n}}_{uv}(z)}{\|\sum_{u,v} \hat{\mathbf{n}}_{uv}(z)\|} \right)^2$ 
18:    Calculate next depth (using Brent's method)
19:     $z_{uv}(s, t) \leftarrow z \leftarrow \text{Brent}(J, z, z_{\min}, z_{\max})$ 
20:  end while
21: end for
22: return  $z_{uv}(s, t), \mathbf{n}_{uv}(s, t), J_{uv}(s, t)$ 

```

one does not always achieve the desired accuracy with this kind of depth-based regularization. In contrast, deflectometry measures slopes, or rather surface normals, with precision several orders of magnitude higher than the depth, but requires information about surface points for regularization [51]. It is therefore useful not to rely solely on depth estimation. Instead, the dense deflectometric measurements of the surface normals can be fused with the various regularization points, which may also be only sparsely available. In doing so, an optimal surface is found whose normals coincide with the deflectometrically measured ones and which has a minimal distance to the calculated regularization points at the same time.

7.2.1 Surface Reconstruction by Depth and Normal Fusion

Starting from a single known surface point, the surface can be integrated from the normal field [100]. However, classical region-growing approaches propagate both the measurement and discretization error along the integration path [50]. In addition, a major challenge is that typically in practical situations the normal field is corrupted by noise and is therefore almost never integrable and curl-free. Due to this, variational approaches are often used where only the integrable part of the normal field is considered and the integration task is formulated as a minimization problem [164]. The general approach of normal field integration can be formulated as an optimization problem as follows: Find the set of surface points $\mathbf{s} \in \mathcal{S}$ for which the functional $E : \mathcal{S} \rightarrow \mathbb{R}$

$$E(\mathbf{s}) = \int_{\mathcal{S}} \|\mathbf{n} - \mathbf{n}_m(\mathbf{s})\|^2 d\sigma \quad (7.17)$$

with surface element $d\sigma$ and surface normal \mathbf{n} takes a global minimum. That said, since in deflectometry the measured normal $\mathbf{n}_m(\mathbf{s})$ depends on the surface \mathbf{s} itself, there exist infinitely many solutions that minimize the above functional [9]. To find the true surface from the infinite manifold of surfaces, regularization points have to be included in the optimization. The surface reconstruction can again be modeled by energy minimization:

$$\arg \min_{\mathbf{s}, \mathbf{n}} \int_{\mathcal{S}} \|\mathbf{n} - \mathbf{n}_m(\mathbf{s})\|^2 + \sum_i \|\mathbf{s} - \mathbf{s}_i\|^2 d\sigma. \quad (7.18)$$

Thus, the searched surface should have minimal distance to the regularization points s_i and at the same time the difference of the surface normals to deflectometrically measured normals \mathbf{n}_m should be minimized. This resolves the ambiguity of deflectometry and results in an overall more robust result for the 3D reconstruction. Though, for practical implementation, the functional needs to be discretized and adapted to the available data. For the light field data available in this work, the depth and normal measurements are located pixel-wise on a discrete grid and the different perspectives of the light field camera are very close to each other. Hence, it is not necessary to search for a general solution of the functional (7.18) in an unconstrained 3D space. Instead, the deflectometric surface reconstruction is formulated here as a discrete gradient integration, and the measured surface points are projected onto a depth map $z(s, t)$. In addition, the corresponding surface gradient $\mathbf{g}(z)$ of this depth map is calculated from the depth-dependent normal field $\mathbf{n}(z)$. Depending on whether perspective or orthographic projection is used different formulas have to be used for this calculation, cf. Sec. 2.4

Since (7.17) is an ill-posed problem, minimization would not yield a meaningful result. By adding additional regularization points a unique solution can be found, but since the coupling between the normal and the surface points is rather weak, and since the regularization points may also only be sparsely available, it makes sense to make further regularizing assumptions to simplify the optimization [8]. In many areas of image processing, *Total Variation* (TV) is used as a popular regularization method because it can handle discontinuities in the data while smoothing noisy measurements [33]. However, it has the disadvantage that linear changes in an intensity profile can form unwanted staircase-like structures after optimization. In depth maps, such intensity changes correspond to a change in depth, e.g., tilted planes, which are by no means uncommon. Therefore, in the field of 3D reconstruction, the TV has the serious disadvantage that such surfaces cannot be reconstructed correctly. In contrast to TV, *Total Generalized Variation* (TGV) avoids this effect by allowing higher-order solutions [28].

Thus, the continuous functional (7.18) is first discretized and the TGV is used as an additional regularization. And similar to the TGV-based image fusion of Pock *et al.* [157] and the normal fusion of Antensteiner *et al.*

[8], a discrete optimization problem that enables a surface reconstruction through a fusion of depth and normal measurements can be defined as

$$\arg \min_{z, \mathbf{g}} \sum_i w_i \|z - z_i\|^2 + w_m \|\mathbf{g} - \mathbf{g}_m(z)\|^2 + \text{TGV}_\alpha^2(z, \mathbf{g}). \quad (7.19)$$

Here, z_i corresponds to any regularizing depth estimates, $\mathbf{g}_m(z)$ calculates the gradient for given depth-dependent normal estimates, w_i and w_m are weights, and z and \mathbf{g} are the sought surface and surface gradient, respectively. Further, the TGV term can be expressed using the gradient operator ∇ and a symmetrized derivative operator $\mathcal{E} = \frac{1}{2}(\nabla + \nabla^T)$:

$$\text{TGV}_\alpha^2(z, \mathbf{g}) = \alpha_1 \|\nabla z - \mathbf{g}\|_1 + \alpha_0 \|\mathcal{E} \mathbf{g}\|_1. \quad (7.20)$$

The purpose of the TGV term is that it strengthens the coupling between the direct estimation of the depth z (respectively surface \mathbf{s}) and the estimation of the surface gradients \mathbf{g} (respectively surface normals \mathbf{n}) by minimizing the distance between the gradient field ∇z calculated from the depth map and the gradient field of the surface \mathbf{g} . In addition, \mathbf{g} is forced by a data term to stay in the proximity of the deflectometrically measured gradient \mathbf{g}_m . At the same time, a deviation of the surface from the depths z_i is penalized. The choice of $\alpha_0 > 0$ causes a smoothing of the gradient field and reduces the influence of noise, and in addition, it implicitly helps to fill holes in the data if gradient information is not available at all locations [28].

Problem (7.19) is convex but discontinuous due to the L_1 -norm. Therefore, as explained in Sec. 2.5, it is necessary to reformulate it as an equivalent convex-concave saddle point problem. This formulation is applied by dualizing only the TGV term and considering the depth and normal data terms as regularization functions

$$G_1(z) = \sum_i w_i \|z - z_i\|^2, \quad G_2(\mathbf{g}) = w_m \|\mathbf{g} - \mathbf{g}_m(z)\|^2. \quad (7.21)$$

The convex conjugate of the weighted L_1 -norms $\alpha_1 \|\cdot\|_1$ and $\alpha_2 \|\cdot\|_1$ contained in the TGV term are calculated to [34]

$$\delta_{Y_1}(\mathbf{y}_1) = \begin{cases} 0, & \|\mathbf{y}_1\|_\infty \leq \alpha_0 \\ \infty, & \|\mathbf{y}_1\|_\infty > \alpha_0 \end{cases}, \quad \delta_{Y_0}(\mathbf{y}_0) = \begin{cases} 0, & \|\mathbf{y}_0\|_\infty \leq \alpha_1 \\ \infty, & \|\mathbf{y}_0\|_\infty > \alpha_1 \end{cases}. \quad (7.22)$$

Algorithm 4 Primal-Dual-Optimization:

Initialize: $z^{(1)} = \bar{z}^1 = \frac{1}{I} \sum_i w_i z_i$, $\mathbf{g}^{(1)} = \bar{\mathbf{g}}^{(1)} = \mathbf{g}_m(z^{(1)})$,
 $\mathbf{y}_1^{(1)} = \mathbf{0}$, $\mathbf{y}_0^{(1)} = \mathcal{E} \mathbf{g}^{(1)}$

- 1: **for** $n = 1, 2, 3, \dots, n_{\max}$ **do**
- 2: Proximal gradient ascent in the dual variables
- 3: $\mathbf{y}_1^{(n+1)} = \text{prox}_{\delta_{Y_1}} \left(\mathbf{y}_1^{(n)} + \tau_{\mathbf{y}_1} \left(\nabla \bar{z}^{(n)} - \bar{\mathbf{g}}^{(n)} \right) \right)$
- 4: $\mathbf{y}_0^{(n+1)} = \text{prox}_{\delta_{Y_0}} \left(\mathbf{y}_0^{(n)} + \tau_{\mathbf{y}_0} \mathcal{E} \bar{\mathbf{g}}^{(n)} \right)$
- 5: Update deflectometric surface gradient
- 6: $\mathbf{g}_m \leftarrow \mathbf{g}_m(z^{(n)})$
- 7: Proximal gradient descent in the primal variables
- 8: $z^{(n+1)} = \text{prox}_{G_1} \left(z^{(n)} - \tau_z \text{div}_{\nabla} \mathbf{y}_1^{(n+1)} \right)$
- 9: $\mathbf{g}^{(n+1)} = \text{prox}_{G_2} \left(\mathbf{g}^{(n)} - \tau_{\mathbf{g}} \left(\text{div}_{\mathcal{E}} \mathbf{y}_0^{(n+1)} - \mathbf{y}_1^{(n+1)} \right) \right)$
- 10: Extrapolation
- 11: $\bar{z}^{(n+1)} = 2z^{(n+1)} - z^{(n)}$
- 12: $\bar{\mathbf{g}}^{(n+1)} = 2\mathbf{g}^{(n+1)} - \mathbf{g}^{(n)}$
- 13: **end for**

At last, with the help of the dual variables $\mathbf{y}_1, \mathbf{y}_0$ the discrete saddle point problem can be formulated as

$$\min_{z, \mathbf{g}} \max_{\mathbf{y}_1, \mathbf{y}_0} \langle \nabla z - \mathbf{g}, \mathbf{y}_1 \rangle + \langle \mathcal{E} \mathbf{g}, \mathbf{y}_0 \rangle + G_1(z) + G_2(\mathbf{g}) - \delta_{Y_1}(\mathbf{y}_1) - \delta_{Y_0}(\mathbf{y}_0). \quad (7.23)$$

The individual variables are scalar, vector or tensor fields parameterized by the spatial pixel grid $w_i, w_m, z, z_i \in \mathbb{R}^{N_s \times N_t}$, $\mathbf{g}, \mathbf{g}_m, \mathbf{y}_1 \in \mathbb{R}^{2 \times N_s \times N_t}$, $\mathbf{y}_0 \in \mathbb{R}^{2 \times 2 \times N_s \times N_t}$, or scalar weighting factors $\alpha_0, \alpha_1 \in \mathbb{R}$.

Using the divergence operators $\text{div}_{\nabla}, \text{div}_{\mathcal{E}}$ that are adjoint to ∇, \mathcal{E} [27], the optimization of the saddle point problem can be solved by iterative gradient descent in the primal variables z, \mathbf{g} and gradient ascent in the dual variables $\mathbf{y}_1, \mathbf{y}_0$ [34]. And, as explained in Sec. 2.5, a corresponding primal-dual optimization scheme can be derived. Alg. 4 shows the optimization algorithm.

Since the deflectometrically measured normals depend on the distance to the surface, the measured gradient field $\mathbf{g}_m(z)$ is updated in each

iteration. The proximal operators can be derived by solving the separate problem (2.38) and can be stated in closed form [34]:

$$\begin{aligned} \text{prox}_{\delta_{Y_1}}(\tilde{\mathbf{y}}_1) &= \frac{\tilde{\mathbf{y}}_1}{\max\left(1, \frac{|\tilde{\mathbf{y}}_1|}{\alpha_1}\right)}, \quad \text{prox}_{G_1}(\tilde{z}) = \frac{\tilde{z} + 2\tau_z \sum_i w_i z_i}{1 + 2\tau_z \sum_i w_i}, \\ \text{prox}_{\delta_{Y_0}}(\tilde{\mathbf{y}}_0) &= \frac{\tilde{\mathbf{y}}_0}{\max\left(1, \frac{|\tilde{\mathbf{y}}_0|}{\alpha_0}\right)}, \quad \text{prox}_{G_2}(\tilde{\mathbf{g}}) = \frac{\tilde{\mathbf{g}} + 2\tau_{\mathbf{g}} w_{\mathbf{m}} \mathbf{g}_{\mathbf{m}}}{1 + 2\tau_{\mathbf{g}} w_{\mathbf{m}}}. \end{aligned} \quad (7.24)$$

While the presented reconstruction algorithm is still very general, it can be applied directly to the light field camera data. In principle, two very general approaches can be considered: multi-depth reconstruction and multi-view reconstruction.

7.2.2 Multi-Depth Reconstruction

In the multi-depth approach, different depth maps are used for regularization and are combined to make the initial depth estimate more robust. For light field-based depth estimation, most of the time only the depth for the central SAI is available. Therefore, only the three central depth maps z_i from Sec. 7.1 are used with $i \in \{\text{direct}, \text{indirect}, \text{multi}\}$. As explained in Sec. 2.4, due to the perspective projection occurring in the light field camera, a variable substitution must be performed so that the surface gradient can be determined from the measured surface normals. By transforming the depth maps

$$\bar{z}_i(s, t) := \ln(z_i(s, t)), \quad (7.25)$$

the surface gradient corresponding to this substitute surface can be easily calculated from the deflectometrically measured normal

$$\hat{\mathbf{n}}_{u_c v_c}(z) = \hat{\mathbf{n}}_{u_c v_c}(\exp(\bar{z})) = (n_1, n_2, n_3)^T \quad (7.26)$$

as a function of the given depth:

$$\mathbf{g}_{\mathbf{m}}(z(s, t)) := -\left(\frac{n_1}{(s-c_s)n_1+(t-c_t)n_2+f_s n_3}, \frac{n_2}{(s-c_s)n_1+(t-c_t)n_2+f_t n_3}\right)^T, \quad (7.27)$$

where the normal is obtained from (7.15). In order to model the perspective projection, the intrinsic camera parameters $c_s := c_s(u_c)$, $c_t := c_t(v_c)$, f_s , f_t from Sec. 6.2.3 are required as well.

Since most depth estimation algorithms provide a confidence measure, this can directly be used as weighting factor w_{direct} and w_{indirect} . And the inverse of the normal disparity is used to calculate w_{multi} . Also, because the deflectometric normal estimation is several magnitudes more accurate than the depth estimation, w_m is selected to be about 100 times larger than the average of the other weights. After finding a minimum for (7.19), the true surface can be derived by back-substitution from (7.25) to $z = \exp(\bar{z})$.

7.2.3 Multi-View Reconstruction

A disadvantage of the naive multi-depth approach is that only the depth estimate for the central SAI is considered, although all other SAIs could also contribute to the reconstruction of the surface. Consequently, the lateral resolution of the reconstruction is limited by the spatial resolution of the central SAI. Furthermore, the depth estimation-based regularization approaches are only applicable to a very limited group of surfaces. In contrast, multi-view regularization through normal disparity minimization can be applied to more diverse surface types and provides regularization information in each SAI.

The individual depth maps $z_{uv}(s, t)$ are initially defined on different virtual sensor planes, therefore they have to be transformed into a common grid. To use the multi-view information to increase the lateral resolution of the reconstructed surface, the individual depth estimates are transformed into a new grid, which does not need to be limited by the spatial resolution of the central SAI. In this case, the perspective projection does not need to be modeled and instead, the grid can be defined by an orthographic projection. Consequently, all depth maps $z_{uv}(s, t)$ are transformed to point clouds s_{uv} using (6.26) and are then orthographically projected onto a new common grid $z(\tilde{s}, \tilde{t})$, where the grid should be designed to enclose all relevant surface points. Alternatively, multi-view regularization could be performed directly on a pre-defined orthographic grid. That is, instead of minimizing the normal disparity for each camera pixel, the disparity for each grid point can be optimized along the depth.

During the optimization, the surface normals corresponding to each depth value are obtained by transforming the depth map back to a point cloud, calculating the normal estimate for each SAI using (7.15), and by taking the average over all estimates. Because an orthographic projection

is used no variable substitution needs to be performed. The surface gradient can be calculated from the deflectometrically measured normal estimate $\hat{\mathbf{n}}(z) = (n_1, n_2, n_3)^T$ as a function of the given depth:

$$\mathbf{g}_m(z(\tilde{s}, \tilde{t})) := -\frac{1}{n_3} \begin{pmatrix} n_1 \\ n_2 \end{pmatrix}. \quad (7.28)$$

For the weight factors, the inverse of the normal disparity is used to define w_i , and the weight of the surface gradients w_m is selected to be about 100 times larger considering that its accuracy is higher as well.

7.3 Evaluation

The next sections examine the steps necessary for specular surface reconstruction and analyze the presented procedures. The experimental setup that was used to conduct the deflectometric measurement is shown in figure 5.17. A 27" monitor with a resolution of 2560×1440 px and a pixel pitch of $233 \mu\text{m}$ was used to display the necessary phase-shift patterns. For image acquisition, the Lytro Illum light field camera was employed.

Since the light field camera can be interpreted as a multi-camera array, a multi-view approach for specular surface reconstruction is pursued in this work. The measurement setup was therefore designed to provide the most ideal conditions for this measurement principle. For the multi-view-based regularization to find a distinct minimum in the normal disparity, the normal field must exhibit substantial variability. According to the findings of Werling [223], this can be achieved with small camera-to-object and monitor-to-object distances, and an angle between camera/monitor axis and mean surface normal of about 45° . The monitor and camera are therefore tilted 90° to each other, and the specular objects are placed at a distance of about 30 to 60 cm in the camera's field of view. In deflectometry, the choice of the focal plane influences the reconstruction. If the camera focuses on the surface, its lateral resolution is maximized, but the monitor is blurred, which increases the uncertainty of the reference feature and leads to a less favorable estimation of the surface normal. When focusing on the monitor, the slope estimate is ideal, but surface features are blurred, which degrades the effective lateral resolution of the reconstruction [223]. As a compromise, in the experimental setup of

this work, the camera is focused on an area slightly behind the surface. Nevertheless, since the Lytro Illum is an unfocused plenoptic camera, the choice of the focal plane is relatively insignificant, since the camera's depth of field is very high.

Furthermore, phase-shift coding was used to obtain reference features, with $M = 12$ shifts and frequencies $\mathbf{f} = (1, 4, 16, 64)$. The probabilistic approach from Ch. 4 was used for phase unwrapping unless specified otherwise. The light field camera was calibrated using the methods from Ch. 6. Hence, for each deflectometric measurement a light field containing the encoded monitor data is retrieved, where the light field resolution is set to $(N_u, N_v, N_s, N_t) = (13, 13, 434, 625)$. The extrinsic calibration of the measurement system was conducted using the calibrated light field camera and the methods from Sec. 5.4.

For the analysis of the reconstruction accuracy, different reference samples were examined. Because their shape is known, this can be used to evaluate the reconstruction accuracy of the presented methods by calculating the distance between the reconstructed surface z and the true surface z_{GT} . For this, the true surface is first fitted onto the reconstructed data and then the depth values are compared. Two error metrics are used: the root-mean-square error and the peak-to-valley ratio

$$\text{RMSE} = \sqrt{\text{Mean}(|z - z_{\text{GT}}|^2)}, \quad (7.29)$$

$$\text{PV} = |\max(z - z_{\text{GT}}) - \min(z - z_{\text{GT}})|, \quad (7.30)$$

where both metrics are calculated over all valid surface points.

7.3.1 Regularization

A partially specular surface is necessary for the evaluation of depth estimation-based regularization. For this purpose, a disk from a hard drive was used as a reference sample, which shows partially reflective areas in the form of color markings and scratches. For the presented regularization methods, the surface must be coded by structured illumination. This allows not only to estimate the monitor coordinates but also to obtain the associated coordinate uncertainty. Since the uncertainty increases dramatically for non-specular or weakly reflective areas, this

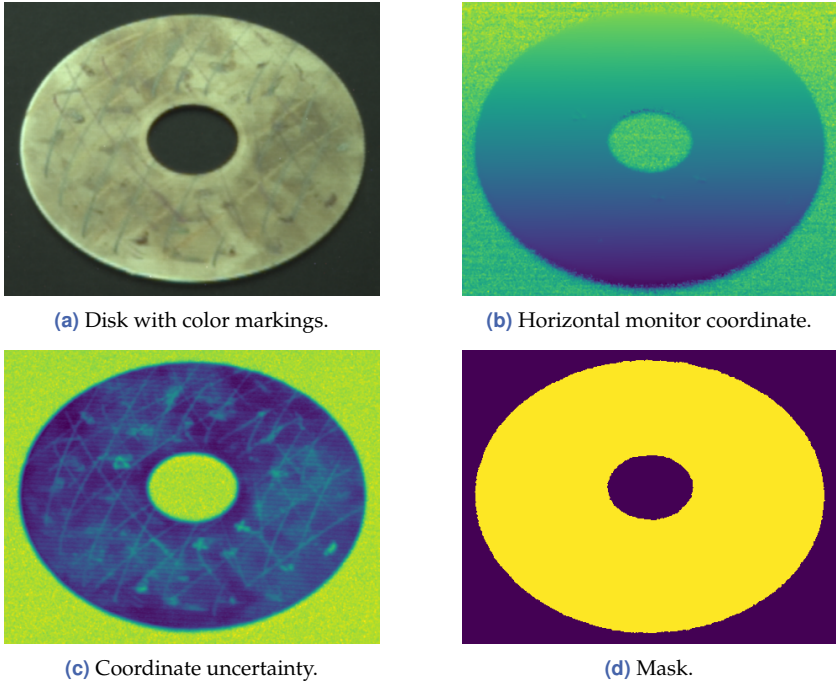


Figure 7.4 Partially specular disk: The mask is calculated by thresholding the uncertainty estimation. The uncertainty increases near scratches and color markings.

can be used as an indicator for the relevant surface areas. Therefore, a threshold on the uncertainty provides masking of the data.

Figure 7.4 shows the disk, the measured vertical monitor coordinates, the coordinate uncertainty, and the resulting calculated mask. Even non-specular components of the background provide registration data. However, these points can easily be removed by using the masking. As expected, the uncertainty is larger for the diffuse components of the surface than for the completely specular ones, but it is still much smaller than the areas outside the disk. Thus, the reconstruction of the specular surface is performed only for those pixels that observe the disk.

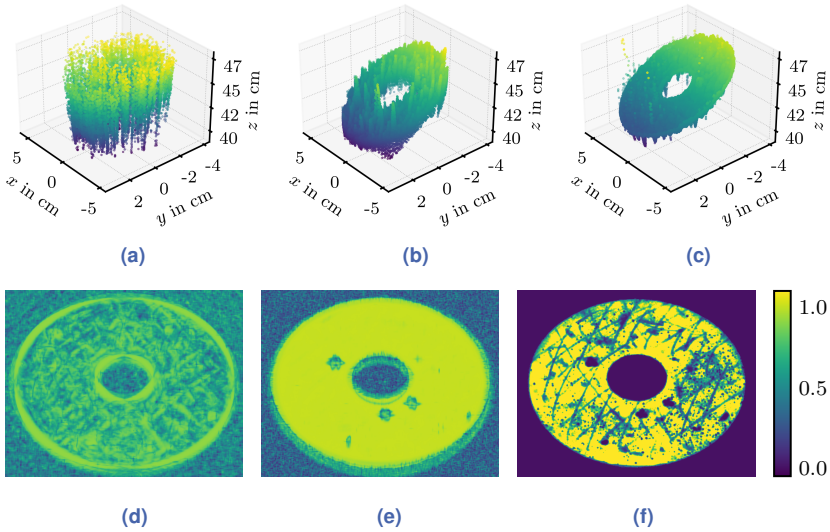


Figure 7.5 Depth estimation and corresponding confidence measures. (a) & (d) Direct depth estimation. (b) & (e) Indirect depth estimation. (c) & (f) Multi-view regularization.

7.3.1.1 Depth estimation

For the direct depth to be measured, no structured illumination is necessary, and instead, the monitor has been turned completely white for adequate brightness. Due to the roughness of the mirror and the color markings, a classical structure tensor-based orientation estimator was used for the depth estimation [218]. This approach provides the disparity, *i.e.*, the slope of the lines, which can be converted into the distance to the disk. Further, it also yields an additional confidence measure for the estimated depth. The confidence is high in the vicinity of structured image areas where the lines in the EPIs are visible. If there is no structure, there are no lines in the EPIs, which results in a low confidence. For the indirect depth estimation, phase-shift coding was used to assign the horizontal and vertical monitor coordinates to each light field pixel. The same algorithm can be used for indirect depth estimation. The only difference is that there are only two “color channels”. Since the method does not perform correctly near strong curvature regions, second-order gradients

are calculated on the registration data. A final confidence measure is then obtained by combining the confidence of the depth with the inverse of the calculated curvature. After estimating the monitor's depth, the indirect depth can easily be calculated by using (7.5). Figure 7.5 shows the estimates of the surface as a point cloud using the different methods as well as the corresponding confidence measures, which are used as weighting for the subsequent surface reconstruction. For comparison, the multi-view regularization is shown as well, where the inverse normal disparity is used as a confidence measure. The figure shows that the direct depth estimation is very noisy because the surface itself has only a few areas with structure. This can also be seen in the corresponding confidence map, where only the areas near the color markings and the edge of the disk show high confidence. The indirect depth estimation is much less noisy since phase-shift coding suppresses image noise. The confidence map is also much more consistent. Yet, the confidence decreases in the vicinity of dents on the surface, as the curvature increases here. The multi-view depth estimate looks the best. The associated confidence values are higher on the fully specular areas than near the color markings. Outside the specular disk, it is zero since these areas are not examined due to the masking.

The major disadvantage of direct and indirect depth estimation is that it only works for very specific surfaces. If the surface is completely specular, no direct surface features can be detected. If the surface has curvature, the depth of the reflection is compressed or stretched. Figure 7.6 shows this behavior for the reconstruction of a convex mirror. While the multi-view regularization can reconstruct the surface, the indirect depth estimation fails completely, even though the surface has only a very small curvature with $\kappa = 1/800 \text{ mm}^{-1}$. The position of the surface is in some cases even estimated to lie behind the camera. In conclusion, the indirect depth estimation may only be used for planar surfaces or needs further improvements. Therefore, for the time being, it should be considered only as a theoretical concept and interesting approach and should be handled with caution for practical use.

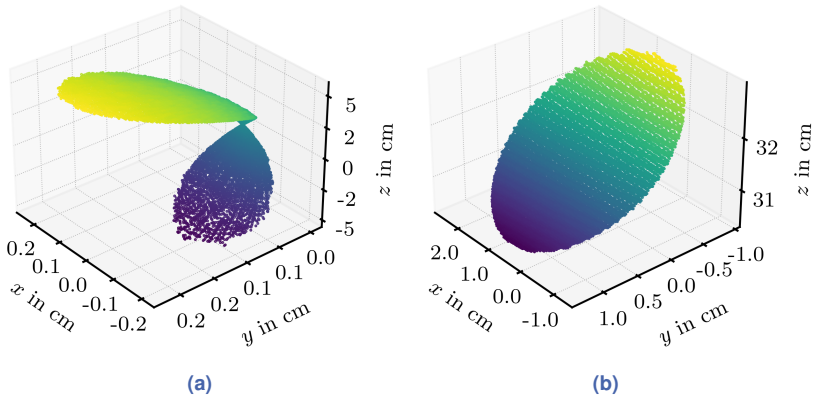


Figure 7.6 Reconstruction of a convex surface: (a) The indirect depth estimation fails even for surfaces with only marginal curvature. (b) The multi-view regularization correctly estimates the surface depth.

7.3.1.2 Normal Disparity Minimization

Minimizing the normal disparity requires neither diffuse surface features nor triangulating the distance to the monitor. Instead, an arbitrarily shaped surface can be found by triangulating the normal field. The inspection of a planar surface and a concave surfaces are shown in figure 7.7. The figure shows the reconstruction of the disparity of a camera pixel as a function of the distance to the surface.

For both surfaces, the disparity increases strongly for decreasing distances, so that the lower bound of the optimization problem (7.16) can be defined without problems. The disparity of the planar surface shows a clear minimum, and it can be seen that the disparity decreases as the distance approaches infinity. It has a local maximum at a distance of about 60 cm. The upper bound for the optimization can therefore be set very loosely since the measurement space of the experimental setup is only slightly larger than 60 cm. The true minimum can therefore be found easily. For the concave surface, two dominant minima emerge. As already explained in Sec. 7.1.2, this is a peculiarity of concave surfaces, such that for stereo deflectometry there are surfaces where the disparity shows equivalent minima at different distances. Fortunately, this is not the case

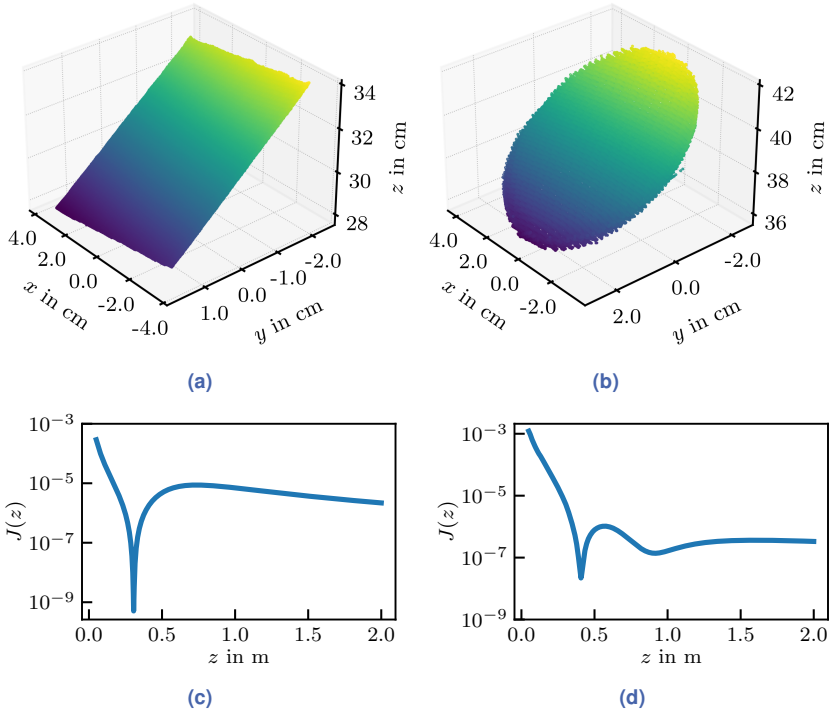


Figure 7.7 Multi-view regularization: The plots show the reconstructed point cloud and the normal disparity $J(z)$ as a function over the distance z . (a) & (c) Planar surface and the disparity of a pixel. (b) & (d) Concave surface and the disparity of a pixel.

for the multi-stereo approach investigated here, and a clear minimum can still be seen. However, it is much more difficult to define the upper limit of the search space, since the disparity of the investigated pixel shows a local maximum at a value of just over 55 cm. There is only a distance of less than 15 cm to the true minimum. Depending on how strongly the surface is inclined, the disparity curve for some pixels is thus shifted further to the right or left. In the worst case, the minimization wanders for some points into the second minimum. However, since the associated disparity is much larger than the one from the true minimum, these erroneous estimates can still be eliminated in post-processing.

The normal disparity can be interpreted as the variance of the angle between the normal estimates. The square root of the disparity thus gives information on how large the spread of the angles is in the point under consideration. For the plane mirror, the minimum of the square root of the disparity is $\sqrt{J} = 22 \mu\text{rad}$ and the local maximum is $\sqrt{J} = 3 \text{ mrad}$. For the concave mirror, the minimum is $\sqrt{J} = 150 \mu\text{rad}$ and the local maximum is $\sqrt{J} = 1 \text{ mrad}$. These very small values are due to the very small baseline between the SAIs. In a standard stereo-deflectometry system with the same baseline, the same disparities would be technically indistinguishable because they would be superimposed by noise. This would make reconstruction impossible [221]. The light field-based multi-view approach with 13×13 SAIs can still resolve the small disparity range despite the small baseline because the multiple views allow a reliable disparity estimation.

While minimizing the disparity already yields surface points, the resulting surface is still not perfect. This is because the triangulation of the normal field, like other triangulation methods, depends on the effective stereo baseline and the distance to the surface, where the uncertainty of the depth estimate increases quadratically with the depth [72]. Thus, for better reconstruction, the normal measurement should be used.

7.3.2 Multi-Depth Reconstruction

To demonstrate the principle approach and the advantages of the different depth estimations, the analysis will be performed here only for the central SAI. Figure 7.8 shows the measurement of the partially specular hard disk and the result of the 3D reconstruction. Because hard disks in general have high planarity, the deviation from the ideal plane is calculated as a quality measure.

The left side of the figure shows the reconstruction error for which the respective regularization result from figure 7.5 was used. The confidence of the depth is used to mask invalid pixels. The pure light field depth estimate of the diffuse surface is therefore only sparsely available in areas of high roughness or near the color markers. All other regions are evaluated as invalid by the depth estimation, which is accounted for by $w_i = 0$ in the fusion. The reconstruction error of the pure regularization is relatively high with an RMSE of 14.80 mm. The indirect depth estimation

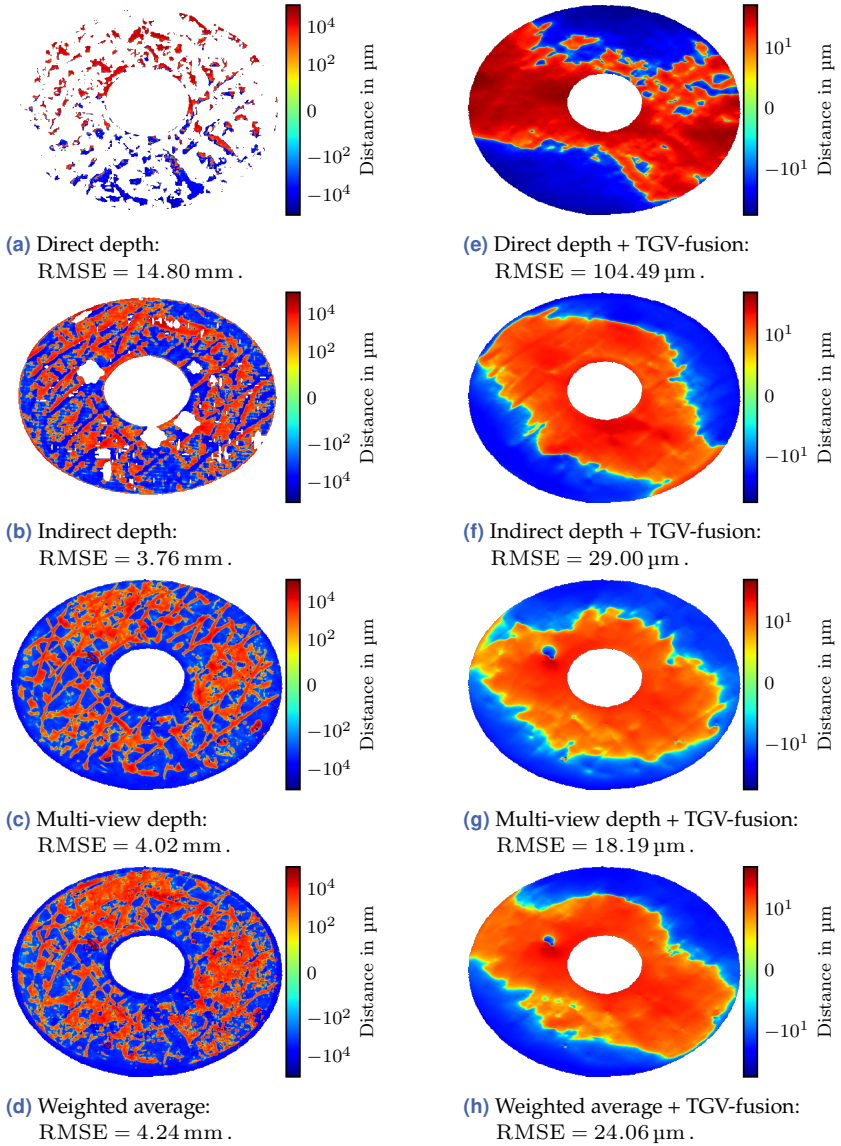


Figure 7.8 Reconstruction of a partially specular disk. The plots show the distance between the disk and an ideal plane. A logarithmic colormap is used for better visualization.

and the multi-view regularization are much denser and less noisy. The RMSE values of the reconstruction are smaller at 3.76 mm and 4.02 mm, respectively. In areas of weak reflection, pixels are marked as invalid in the indirect depth estimation because the confidence is very low and the depth estimation yields significantly erroneous values. The multi-view estimation also has small confidence values in the same areas, but can still provide reasonably correct values.

If the regularization is used to provide support points for the normal integration from Sec. 7.2.1, then the reconstruction error can be significantly reduced. Because the depth estimation is only available sparsely in some places, the intermediate values must be interpolated as initialization of the surface, with the help of which the surface normals can then be calculated. For all further steps in Alg. 4 no interpolation has to be done, because it is sufficient to use only the valid pixels as support points for the fusion. The right side of figure 7.8 shows the corresponding results of the fusion. While the direct depth estimation has a relatively high error, the multiple regularization points are sufficient to allow a reasonably good reconstruction. Interestingly, the reconstruction with multi-view regularization with $\text{RMSE} = 18.19 \mu\text{m}$ is better than the one with the indirect depth estimation with $\text{RMSE} = 29.00 \mu\text{m}$, although the regularization points of the indirect depth estimation have the smallest error overall. This can be explained by the fact that although the disk has been manufactured precisely and with a high degree of flatness, it may have a very slight curvature due to external forces, *e.g.*, resulting from adding paint markings or from the deliberate application of scratches and dents. Therefore, the indirect regularization yields slightly incorrect data, as explained before.

Since all regularization methods use different information as a basis, the regularization points have different uncertainties and can thus jointly contribute to the improvement of the reconstruction. For this purpose, firstly, a weighted average of the individual regularization is calculated. Further, in the depth and normal fusion (7.19) all depth estimates are used jointly, weighted by the respective confidences. Figure 7.8(d) and (h) show the respective reconstruction errors. Although it often helps to merge different sources of information, in this example the result both times is worse than when using only the multi-view regularization. The

cause of this may be that the used confidence measures do not necessarily represent the uncertainty of the regularization and therefore may not be used as equivalent weights. On the other hand, the regularization methods may show systematic errors that cannot be assessed using a confidence estimation.

7.3.3 Multi-View Reconstruction

The light field depth estimation algorithms proposed in the literature generally provide only the depth of the central SAI, because the high redundancy of the light field is usually not needed after the estimation. In principle, the algorithms could be adapted to compute the depths in other SAIs, but the depth estimation-based regularization approaches had other drawbacks, as noted in the last sections, so they will not be considered any further here. The advantage of multi-view regularization is that multiple views can be used to increase the lateral resolution of the reconstruction. For this purpose, all depth estimates from all SAIs are transformed into a uniform grid.

7.3.3.1 Evaluation of the Reconstruction Accuracy

Since the hard disk from the previous sections can only be regarded as approximately planar, different reference mirrors with known shapes are used to quantify the accuracy of the reconstruction in the following.

For the first experiment, a precision surface mirror with $\lambda/20$ flatness is used as the surface under test. With the reference wavelength of 632.8 nm, the mirror has a maximum peak-to-valley deviation from the perfect plane of 31.64 nm. Thus, compared to the achievable accuracy of the measurement system in this work, it can be considered absolutely flat. Therefore, as a quality measure, a perfect plane is fitted into the reconstructed point cloud, and for each point, the distance to this plane is evaluated as a quality measure. Figure 7.9 shows the results of the surface reconstruction. The point cloud, which can be obtained with the help of multi-view regularization, already provides a reasonably good reconstruction. Overall, however, the reconstructed surface still appears slightly noisy. The corresponding error map indicates that the surface is not yet smooth. After optimization by fusion with the estimated surface normals, the result

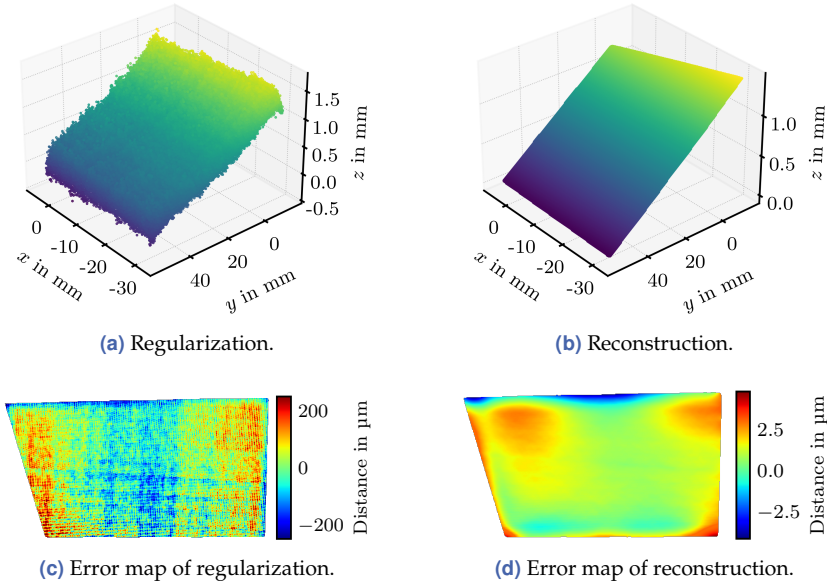


Figure 7.9 Reconstruction of a planar mirror: (a) & (c) The accuracy of the regularization is quantified with $\text{RMSE} = 89.71 \mu\text{m}$, $\text{PV} = 478.24 \mu\text{m}$. (b) & (d) The accuracy of the reconstruction is quantified with $\text{RMSE} = 0.99 \mu\text{m}$, $\text{PV} = 7.94 \mu\text{m}$.

is better. The RMSE decreases to $0.99 \mu\text{m}$ and the PV metric yields $7.94 \mu\text{m}$. Thus, the reconstruction result shows comparable accuracy to other deflectometric measurement systems from the literature [103, 154, 230].

In a second experiment, a convex surface is to be reconstructed. The reference mirror has a radius of curvature of $R = 1/\kappa = 800 \text{ mm}$ and planarity of $\lambda/2$, which can still be considered a nearly perfect reference for the measurement accuracy of the deflectometry system used in this work. Since the shape of the mirror is known, the distance to the ideal surface is again used as a quality measure. Figure 7.10 shows the results of the surface reconstruction. The point cloud of the regularization appears very noisy and there are strong errors at the edge of the surface. This can also be seen in the corresponding error map. The overall error is quite high with $\text{RMSE} = 333.85 \mu\text{m}$ and $\text{PV} = 1.50 \text{ mm}$. Looking at the surface in detail, a systematic wave-like structure can be seen on the surface. An

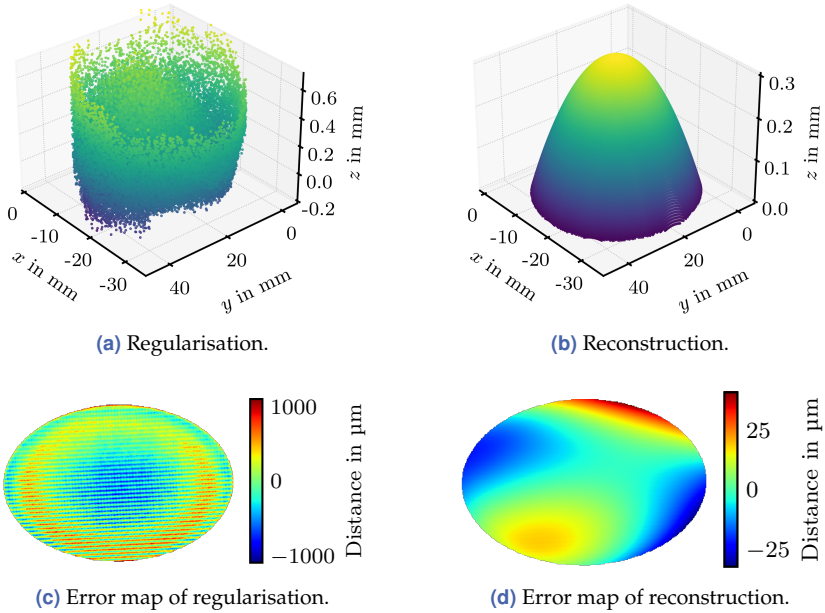


Figure 7.10 Reconstruction of a convex mirror: (a) & (c) The regularization results in $\text{RMSE} = 333.85 \mu\text{m}$, $\text{PV} = 1.50 \text{ mm}$. (b) & (d) The reconstruction results in $\text{RMSE} = 12.02 \mu\text{m}$, $\text{PV} = 41.03 \mu\text{m}$.

explanation for this effect could be vibrations during the measurement or a slightly faulty calibration. However, an exact cause is not known. Still, the reconstruction of the surface using the depth and normal fusion shows reasonable good results and the ripples in the surface disappear as well. The shape of the surface is clearly recognizable and the accuracy increases strongly to $\text{RMSE} = 12.02 \mu\text{m}$ and $\text{PV} = 41.03 \mu\text{m}$. However, the reconstruction accuracy is not as good as for the planar surface, which is probably due to the inferior result of the regularization.

As a last experiment, a concave surface is reconstructed. The reference mirror has a radius of curvature of $R = 406 \text{ mm}$ and planarity of $\lambda/4$. Figure 7.11 shows the results of the surface reconstruction. The surface can already be recognized in the point cloud of the regularization. As before, a wave-like structure appears on the surface. The error of the reg-

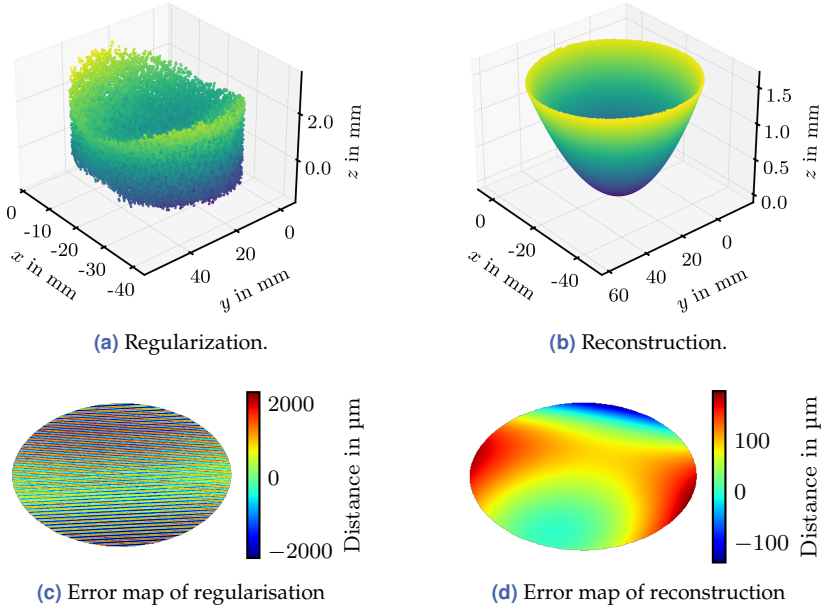


Figure 7.11 Reconstruction of a concave mirror: (a) & (c) The regularization results in $\text{RMSE} = 1.34 \text{ mm}$, $\text{PV} = 4.00 \text{ mm}$. (b) & (d) The reconstruction results in $\text{RMSE} = 54.75 \mu\text{m}$, $\text{PV} = 210.50 \mu\text{m}$.

ularization is relatively high with $\text{RMSE} = 1.34 \text{ mm}$ and $\text{PV} = 4.00 \text{ mm}$, which can also be attributed to the peculiarities of concave surfaces. As explained in Sec. 7.3.1.2, the disparity minimization of concave surfaces is more susceptible to noise. Still, the final result of the depth and normal fusion shows a strongly improved result.

7.3.3.2 Lateral Resolution

An advantage of the multi-view regularization is that the lateral resolution of the surface reconstruction is not limited by the spatial resolution of the central SAI. The resolution can be specified by the user. The light field used here has the dimension $(N_u, N_v, N_s, N_t) = (13, 13, 434, 625)$. Assuming that each SAI increases the resolution, the maximum possible resolution of an orthographic grid is therefore approximately 13 times the

resolution of a single SAI. To show the advantage of the higher resolution, the partially specular disk from the previous sections will be examined in the following. The reconstruction was performed with the resolutions 400×400 , 1500×1500 , and 4000×4000 , where the grid is defined to enclose the valid surface points as accurately as possible. The smallest resolution corresponds approximately to the resolution that would be obtained if only the central SAI would be considered in the reconstruction.

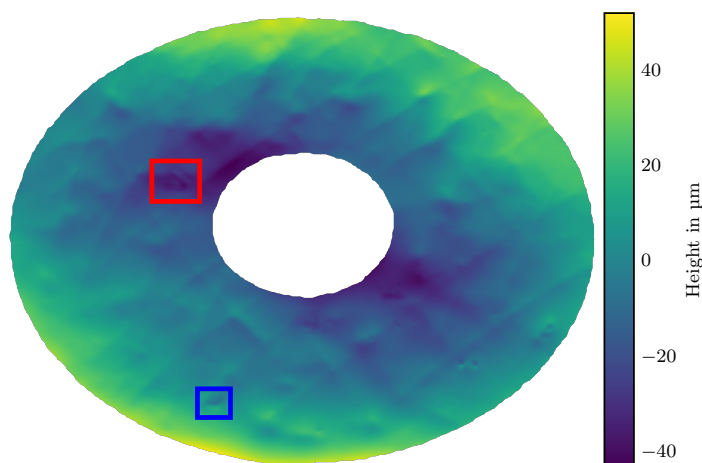
Figure 7.12 shows the results of the reconstruction of the disk, as well as close-up views that have been reconstructed with the different resolutions. The disk shows local defects in the form of scratches and dents. With the low resolution, the defects in the disk can hardly be identified. This shows that it is not sufficient to use only the central SAI for the reconstruction. With a resolution of 1500×1500 , the defects in the disk can be recognized very well. If the resolution is increased even further, there is hardly any noticeable improvement. This is probably related to the fact that the surface normal \mathbf{n} for the reconstruction is calculated as the weighted average of the normal estimates \mathbf{n}_{uv} from all SAIs. A more selective choice of the best normal estimated from all SAIs or a more sophisticated weighting might therefore improve the results.

7.3.4 Influence of the Calibration

A substantial part of this thesis was dedicated to the accurate calibration of the deflectometric measuring system. That the effort was worthwhile will be shown in the following.

For the evaluation, the surface reconstruction was carried out based on four different configurations of the system calibration. The light field camera was calibrated using the procedure of Bok *et al.* [20] and using the generic light field reconstruction procedure presented in Ch. 6. In addition, for each camera calibration, the influence of the monitor model from Sec. 5.3 was analyzed. To assess the reconstruction quality, the planar reference mirror was again used and the deviation from the ideal plane was evaluated. Figure 7.13 shows the results of the respective reconstructions.

The results impressively demonstrate that the camera model has a significant impact on the reconstruction accuracy. With the calibration method by Bok *et al.* [20] the surface can still be reconstructed with high accuracy, but if the proposed generic LF-reconstruction is used, the re-



(a) Reconstructed surface.

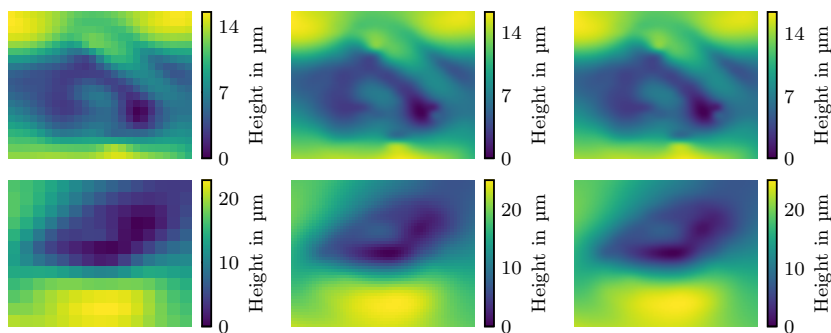
(b) 400×400 .(c) 1500×1500 .(d) 4000×4000 .

Figure 7.12 Reconstruction of the partially specular disk with different resolution of the grid parameters. Local defects can be identified by increasing the lateral resolution. The top row shows the content of the red rectangle with an area of approximately $5 \text{ mm} \times 4 \text{ mm}$. The bottom row shows the content of the blue rectangle with an area of approximately $3 \text{ mm} \times 2.5 \text{ mm}$.

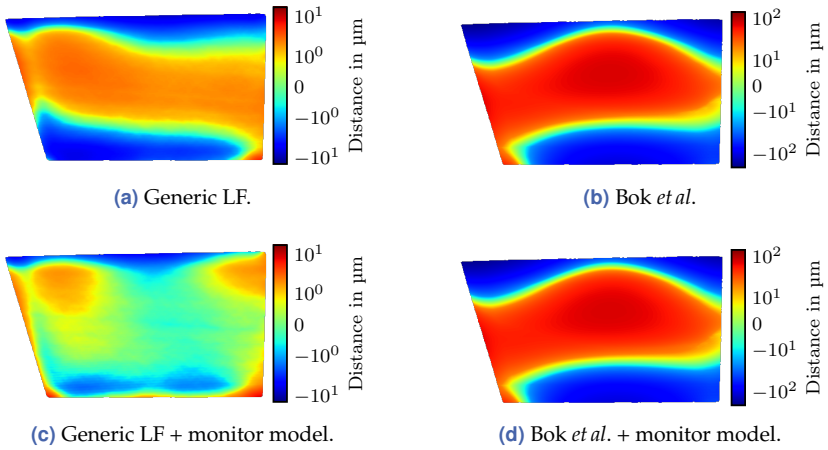


Figure 7.13 Influence of the calibration on the reconstruction accuracy. The plots show the distance to an ideal plane. Note the difference in scale. (a) RMSE = $2.30\ \mu\text{m}$, PV = $8.71\ \mu\text{m}$. (b) RMSE = $50.19\ \mu\text{m}$, PV = $219.19\ \mu\text{m}$. (c) RMSE = $0.99\ \mu\text{m}$, PV = $7.94\ \mu\text{m}$. (d) RMSE = $49.77\ \mu\text{m}$, PV = $216.92\ \mu\text{m}$.

sults are significantly better. For the method of Bok *et al.*, it seems that the surface shows a slight curvature. This is most likely caused by the comparatively inferior geometric calibration. As was pointed out in Sec. 6.3, the quality of the SAIs at the edge of the angular plane is worse than in the center. Accordingly, the calibration error is higher, which corrupts the deflectometric triangulation of the normal field and, in addition, leads to an erroneous normal measurement. The precise calibration of the light field camera presented in this work is therefore indispensable for the deflectometric reconstruction of specular surfaces.

The monitor model also affects the result, although not as much as the camera calibration. For both camera calibrations, the accuracy is slightly better with the monitor model. The RMSE is minimally better and the PV decreases a few micrometers in both cases. Comparing the results of the generic LF-reconstruction, it is noticeable that without using the monitor model, the surface shows a slight curvature. The falsely assumed planar monitor display is transferred into a falsified surface reconstruction. By using a monitor model, this systematic error can be corrected.

7.4 Summary

This chapter described how the special optical properties of a light field camera can be used for the deflectometric reconstruction of specular surfaces. The information contained in the light field opens up the possibility of regularizing the ambiguity of the deflectometric normal estimation. It was explained how classical light field depth estimation algorithms can be used to extract regularizing information and how the light field camera can be interpreted as a highly multi-view camera array to enable a multi-view regularization by triangulating the normal field. To further increase the reconstruction accuracy, the normal measurements were fused with the surface obtained from the regularization using a variational optimization approach, which was solved with a primal-dual optimization algorithm.

Experiments showed that, despite regularization techniques of different quality, good results can be achieved. The depth estimate-based regularization methods are only applicable for a very special group of surfaces, whereas the multi-view regularization can be used for arbitrary specular free-form surfaces and provides better results. Further investigations showed that by fusion of the depth and normal estimates the reconstruction accuracy can be drastically improved so that accuracies in the lower micrometer range become possible, which is comparable to other deflectometry systems from the literature. Moreover, the calibration of the measurement system had a significant influence on the accuracy of the reconstruction. Hence, the calibration methods that were presented in this thesis are very well suited for deflectometry.

In summary, light field-based deflectometry can be realized efficiently and in a compact design. Despite the very small stereo baseline between the SAIs, but due to their immense number, high accuracy of the measurement can be achieved. This enables the reconstruction of the global surface form as well as local defects.

8 Conclusion

This thesis investigated how light field imaging can be efficiently utilized for deflectometry. While the key statements of the individual research topics have already been summarized at the end of the respective chapters, the results achieved with a view on the context of the entire thesis are summarized here.

8.1 Summary

Deflectometry requires structured illumination, where the encoding of the monitor pixel intensities enables the registration of camera pixels to monitor pixels. In this thesis, multi-frequency phase-shifting techniques were used as they provide high measurement accuracies and allow sub-pixel accurate registration. At the same time, however, they introduce ambiguities that can only be resolved using phase unwrapping methods. Furthermore, several classical methods for phase unwrapping have been studied. As a major contribution, a new probabilistic approach for phase unwrapping was proposed. Using circular statistics, both the periodicity of the phase is taken into account and the estimation of the phase uncertainty can be included in the unwrapping process, thus automatically compensating for individual erroneous phase measurements. By performing a maximum-likelihood optimization on the probability distribution of the phase measurement, the optimal monitor coordinate can be decoded for each camera pixel. Moreover, it was shown that by modeling the local pixel neighborhood, the robustness of the method can be improved further, leading to a probabilistic approach for spatio-temporal phase unwrapping. Overall, the results showed that the proposed methods are significantly more robust to noise influences than state-of-the-art methods, resulting in ideal starting conditions for use in deflectometry.

Highly accurate calibration is an important prerequisite for precise deflectometric measurements. In this thesis, a generic camera model was used to calibrate the light field camera, in which the view rays associated with each pixel are estimated individually, resulting in a highly accurate calibration. To estimate the camera parameters, it was proposed to split the calibration into two subproblems, a ray calibration and a pose estimation, and it was shown how an alternating minimization approach can be used to deal with the tremendous number of parameters. Calibration features were obtained using phase-shift coding and the estimated coordinate uncertainty was used as weighting in the optimization. An analytical solution was given for the ray calibration, and the pose was optimized using a gradient descent-based method on the rotation manifold. Since the reference monitor used for calibration is not ideal, the shape and the refraction at the cover glass were modeled, and it was shown how the estimation of the respective parameters could be efficiently integrated into the generic calibration framework. Finally, experiments demonstrated the superiority of the presented generic method over classical calibrations and other generic approaches.

While the generic calibration is very precise, it provides an unconstrained bundle of camera rays. The relationships among these rays are lost. Thus, with the generic camera model, it is extremely difficult to identify to which pixel a 3D point is projected or which ray is closest to that point. For deflectometry, this forward and backward projection is a necessity for a correct surface triangulation. In the case of the generically calibrated light field camera, this means that the 4D information contained in the light field and, in particular, the relations between the individual camera rays must be recovered. To achieve this, this thesis proposed to use the generic camera calibration as a basis to perform a generic light field reconstruction. The approach reconstructs the light field from the camera raw data by only considering the geometry of the camera rays and by resampling the corresponding intensity values. Experiments validated the approach by reconstructing light fields from different light field cameras. A comparison with state-of-the-art light field reconstruction methods showed that the presented method is better able to compensate for lens aberrations since these are already optimally contained in the generic bundle of rays. The method was therefore able

to reconstruct the information of the observed scene as well as to return the geometric structure of the light field with the help of an adequate rectification and calibration. This can be done regardless of whether the light field camera is based on microlenses, mirrors, or coded apertures, or whether it is implemented by using a camera array.

With the help of the registration and calibration, a deflectometric measurement could finally be carried out. Since the deflectometric normal measurement is inherently ambiguous, different regularization methods were proposed, which take advantage of the special properties of the light field camera. As the most important aspect, a multi-view approach was adapted which interprets the light field camera as a highly multiplexed camera array, where possible surface normals can be calculated in the field of view of each of these cameras. The normals differ in general, yet must coincide on the true surface. By comparing the normal fields, an initial estimate of the surface can be found. Moreover, an approach was presented to fuse the regularization points with the deflectometrically measured surface normals to further increase the accuracy of the surface reconstruction. The fusion was formulated as a variational optimization problem and a solution was found using a primal-dual algorithm. Experiments showed that with regularization alone, the mirror surfaces can be reconstructed with accuracies in the upper micrometer range. By fusion of the depth and normal estimates, the result could be drastically improved again, where the reconstructed surface shape deviated from the reference shape with RMSE values around $1\text{ }\mu\text{m}$ and peak-to-valley ratios of less than $10\text{ }\mu\text{m}$. The investigated light field-based deflectometry approach thus comes within similar orders of magnitude as comparable methods from the literature. Furthermore, by evaluating the influence of the system calibration, it became clear that the proposed generic light field reconstruction provides significantly higher surface reconstruction accuracy as compared to when using state-of-the-art light field calibration methods. This showed that the precise calibration presented in this thesis is an imperative necessity for deflectometric reconstruction of specular surfaces.

In conclusion, light field-based deflectometry can be efficiently implemented and enables high-precision reconstruction of specular surfaces.

8.2 Outlook

The following is a presentation of ideas and concepts that have emerged in the context of this thesis and that present future research opportunities.

The generic camera model presented in this thesis represents only the simple geometric properties of the camera, moreover, it assumes that each pixel can be perfectly described by a single ray. In reality, however, the widening of the light ray induced by the camera optics results in not every distance being in focus. Comparatively, a cone would be a more accurate description, where its expansion and shape change as a function of distance. An estimation of the cone parameters would make the generic camera model more complete. When a ray hits the monitor, an intersection plane is created between the corresponding cone and the monitor plane. The area observed by the corresponding pixel is elliptically distorted to different degrees depending on the tilt of the monitor. The uncertainty of the horizontal and vertical monitor coordinates, which can be estimated by phase-shift coding, corresponds to the axes of this ellipse. The observation of different intersection planes could open up the possibility of determining the distance-dependent focus parameters for each ray in addition to the geometric parameters. With multi-focus light field cameras, the proposed generic light field reconstruction is still subject to limitations, since here sharp rays and blurred rays are processed together. The extension of the generic camera model by focus parameters should be helpful for the generic light field reconstruction as well.

While the generic light field reconstruction yields good results for classical light field cameras, it would be interesting to apply it in the context of spectrally coded light field cameras as proposed by Schambach [178]. These cameras encode the spatial dimension of the light field using a spectral mask such that there is only a single spectral channel for each pixel. The geometric calibration of these cameras is difficult since adjacent pixels contain very different information. The generic approach could circumvent these difficulties by reconstructing an individual light field for each spectral channel, which could then be merged into a single calibrated and rectified light field. Whether the SAIs remain spectrally encoded or whether one directly reconstructs the complete light field for each spectral channel, *e.g.*, by using the generic superresolution approach described in this thesis, depends on the requirements of the following applications.

The reconstruction of specular surfaces still has potential for improvement. The lateral resolution of the measurement is not limited by the spatial resolution of the light field, as has been shown, but can be increased by considering the angular dimension. However, this could only increase the lateral resolution to a certain extent, since the surface normals were always calculated as the average of the estimates from all SAIs. Thus, a more sophisticated calculation of the normals could improve the resolution. Alternatively, the depth and normal fusion could be combined with variational superresolution approaches [206].

The light field camera is equivalent to a multiple camera array where the baseline between the cameras is in general not much larger than 1 mm depending on the model. A perspective change in the light field can therefore be considered as a quasi-continuous movement of a single camera. This allows estimating specular flow which occurs when the reflection of a structured environment is observed on a specular surface. While this thesis focused on a high accuracy reconstruction of specular surfaces, specular flow can also be used for defect detection tasks or 3D measurements with lower accuracy requirements [2, 144]. In the context of this thesis, research was conducted on using a light field camera to induce specular flow and use CNNs to reconstruct the surface, where only a structured but unknown environment was required. This thesis does not cover this approach, since it only works with synthetic data to some extent, but cannot be used for real cameras without further modifications. Unlike CNN-based disparity estimation commonly used in the literature, surface reconstruction and depth estimation are strongly coupled to the respective camera parameters. Training the CNN on synthetic data and applying it to real data is therefore not possible for the time being. Other applications have already shown that it is possible to consider the camera parameters during the design of the CNN's architecture [52]. Adopting this approach for the reconstruction of specular surfaces could lead to interesting results. The advantage of specular flow is that it does not require a temporal encoding of the illumination but only needs a structured reference scene. With the light field camera, a single exposure already contains all the required information for specular flow calculation. This would open up the possibility for deflectometry in motion.

9 Appendix

9.1 Calibration

9.1.1 Variables

Matrices of pose subproblem

In Sec. 5.2.5, for every single pose with index k , an optimization problem with objective function

$$f(\mathbf{R}_k, \mathbf{t}_k) = \sum_i w_{ik} \|(\mathbf{R}_k \mathbf{x}_{ik} + \mathbf{t}_k) \times \mathbf{d}_i - \mathbf{m}_i\|^2 \quad (9.1)$$

is obtained. This can be written in a more compact form by using the *Kronecker* identity (2.3), the cross product operator (2.4), the *vec*-operator with $\mathbf{r}_k = \text{vec}(\mathbf{R}_k)$, and the introduction of some new variables:

$$\mathbf{A}_{\text{rr},k} = \sum_i w_{ik} (\mathbf{x}_{ik} \mathbf{x}_{ik}^T) \otimes ([\mathbf{d}_i]_{\times} [\mathbf{d}_i]_{\times}^T), \quad (9.2)$$

$$\mathbf{A}_{\text{tt},k} = \sum_i w_{ik} [\mathbf{d}_i]_{\times} [\mathbf{d}_i]_{\times}^T, \quad (9.3)$$

$$\mathbf{A}_{\text{tr},k} = \sum_i 2w_{ik} [\mathbf{d}_i]_{\times} (\mathbf{x}_{ik}^T \otimes [\mathbf{d}_i]_{\times}^T), \quad (9.4)$$

$$\mathbf{b}_{\text{r},k} = \sum_i -2w_{ik} (\mathbf{x}_{ik}^T \otimes [\mathbf{d}_i]_{\times}^T)^T \mathbf{m}_i, \quad (9.5)$$

$$\mathbf{b}_{\text{t},k} = \sum_i 2w_{ik} [\mathbf{d}_i]_{\times}^T \mathbf{m}_i, \quad (9.6)$$

$$h_k = \sum_i w_{ik} \|\mathbf{m}_i\|^2, \quad (9.7)$$

which results in the more compact form:

$$f(\mathbf{r}_k, \mathbf{t}_k) = \mathbf{r}_k^T \mathbf{A}_{\text{rr},k} \mathbf{r}_k + \mathbf{t}_k^T \mathbf{A}_{\text{tt},k} \mathbf{t}_k + \mathbf{t}_k^T \mathbf{A}_{\text{tr},k} \mathbf{r}_k + \mathbf{b}_{\text{r},k}^T \mathbf{r}_k + \mathbf{b}_{\text{t},k}^T \mathbf{t}_k + h_k. \quad (9.8)$$

Matrices of rotation subproblem

In Sec. 5.2.5, an optimization problem for the rotation estimation with objective function

$$f(\mathbf{R}) = \mathbf{r}^T \mathbf{A} \mathbf{r} + \mathbf{b}^T \mathbf{r} + c \quad (9.9)$$

is obtained. The corresponding parameters can be easily derived by inserting (5.24) in (9.8):

$$\mathbf{A} = \mathbf{A}_{rr} - \frac{1}{4} \mathbf{A}_{tr}^T \mathbf{A}_{tt}^{-1} \mathbf{A}_{tr}, \quad (9.10)$$

$$\mathbf{b} = \mathbf{b}_r - \frac{1}{4} \mathbf{A}_{tr}^T \mathbf{A}_{tt}^{-1} \mathbf{b}_t, \quad (9.11)$$

$$c = h - \frac{1}{4} \mathbf{b}_t^T \mathbf{A}_{tt}^{-1} \mathbf{b}_t. \quad (9.12)$$

9.1.2 Riemannian Gradient and Hessian on $\text{SO}(3)$

In Sec. 5.2 to minimize the rotation subproblem (5.25), the Riemannian Gradient and Hessian were needed. In the following, it is demonstrated how the gradient and Hessian for the rotation subproblem (5.25) from Sec. 5.2 can be calculated.

Gradient

In the $\mathfrak{so}(3)$ -tangent space, the derivative in direction $\boldsymbol{\xi}$ is calculated. With $\mathbf{R}(\boldsymbol{\xi}) = e^{[\boldsymbol{\xi}]_{\times}} \mathbf{R}$, $\mathbf{r} = \text{vec}(\mathbf{R})$ and \mathbf{Z} as defined in (2.7), it follows:

$$D_{\boldsymbol{\xi}} f(\mathbf{R}) = \partial_{\varepsilon} f_{\boldsymbol{\xi}\varepsilon}(\mathbf{R})|_{\varepsilon=0}, \quad (9.13)$$

$$\partial_{\varepsilon} f_{\boldsymbol{\xi}\varepsilon}(\mathbf{R}) = \partial_{\varepsilon} \mathbf{r}(\boldsymbol{\xi}\varepsilon)^T \partial_{\mathbf{r}} f(\mathbf{R})|_{\mathbf{r}=\mathbf{r}(\boldsymbol{\xi}\varepsilon)} = \partial_{\varepsilon} \mathbf{r}(\boldsymbol{\xi}\varepsilon)^T 2(\mathbf{A} \mathbf{r}(\boldsymbol{\xi}\varepsilon) + \mathbf{b}) \quad (9.14)$$

$$= 2 \text{vec}([\boldsymbol{\xi}]_{\times} e^{\varepsilon [\boldsymbol{\xi}]_{\times}} \mathbf{R})^T (\mathbf{A} \mathbf{r}(\boldsymbol{\xi}\varepsilon) + \mathbf{b}). \quad (9.15)$$

With $\varepsilon \rightarrow 0$ it follows:

$$D_{\boldsymbol{\xi}} f(\mathbf{R}) = \partial_{\varepsilon} f_{\boldsymbol{\xi}\varepsilon}(\mathbf{R})|_{\varepsilon=0} = 2 \text{vec}([\boldsymbol{\xi}]_{\times} \mathbf{R})^T (\mathbf{A} \mathbf{r} + \mathbf{b}) \quad (9.16)$$

$$\stackrel{(2.3)}{=} ((\mathbf{R}^T \otimes \mathbf{I}) \text{vec}([\boldsymbol{\xi}]_{\times}))^T (\mathbf{A} \mathbf{r} + \mathbf{b}) \quad (9.17)$$

$$\stackrel{(2.6)}{=} 2 \boldsymbol{\xi}^T \mathbf{Z}^T (\mathbf{R} \otimes \mathbf{I}) (\mathbf{A} \mathbf{r} + \mathbf{b}) = \boldsymbol{\xi}^T \text{grad}(f). \quad (9.18)$$

Finally, the gradient of the locally parameterized objective function can be obtained:

$$\text{grad}(f) = 2\mathbf{Z}^T (\mathbf{R} \otimes \mathbf{I}) (\mathbf{A}\mathbf{r} + \mathbf{b}) . \quad (9.19)$$

Hessian

The second order derivative is calculated similarly to the previous calculations:

$$D_{\boldsymbol{\xi}} \text{grad}(f) = \lim_{\varepsilon \rightarrow 0} \partial_{\varepsilon^2}^2 f_{\varepsilon \boldsymbol{\xi}}(\mathbf{R}) , \quad (9.20)$$

$$\boldsymbol{\xi}^T \partial_{\varepsilon} \text{grad}(f) = \boldsymbol{\xi}^T \partial_{\varepsilon} 2\mathbf{Z}^T (\mathbf{R}(\boldsymbol{\xi}\varepsilon) \otimes \mathbf{I}) (\mathbf{A}\mathbf{r}(\boldsymbol{\xi}\varepsilon) + \mathbf{b}) \quad (9.21)$$

$$= \partial_{\varepsilon} \left(2\text{vec}([\boldsymbol{\xi}]_{\times} e^{\varepsilon[\boldsymbol{\xi}]_{\times}} \mathbf{R})^T (\mathbf{A}\text{vec}(e^{\varepsilon[\boldsymbol{\xi}]_{\times}} \mathbf{R}) + \mathbf{b}) \right) . \quad (9.22)$$

With $\varepsilon \rightarrow 0$ it follows:

$$D_{\boldsymbol{\xi}} \text{grad}(f) = 2\text{vec}([\boldsymbol{\xi}]_{\times}^2 \mathbf{R})^T (\mathbf{A}\mathbf{r} + \mathbf{b}) + 2\text{vec}([\boldsymbol{\xi}]_{\times} \mathbf{R})^T \mathbf{A}\text{vec}([\boldsymbol{\xi}]_{\times} \mathbf{R}) . \quad (9.23)$$

With the reshape operator $\text{mat}(\text{vec}(\mathbf{A})) = \mathbf{A}$, it follows :

$$\begin{aligned} 2\text{vec}([\boldsymbol{\xi}]_{\times}^2 \mathbf{R})^T (\mathbf{A}\mathbf{r} + \mathbf{b}) &= 2\boldsymbol{\xi}^T \mathbf{Z}^T ([\boldsymbol{\xi}]_{\times} \mathbf{R} \otimes \mathbf{I}) (\mathbf{A}\mathbf{r} + \mathbf{b}) \\ &\stackrel{(2.3)}{=} 2\boldsymbol{\xi}^T \mathbf{Z}^T \text{vec}(\text{mat}(\mathbf{A}\mathbf{r} + \mathbf{b}) \mathbf{R}^T [\boldsymbol{\xi}]_{\times}^T) \\ &\stackrel{(2.3)}{=} 2\boldsymbol{\xi}^T \mathbf{Z}^T (\mathbf{I} \otimes \text{mat}(\mathbf{A}\mathbf{r} + \mathbf{b}) \mathbf{R}^T) \text{vec}([\boldsymbol{\xi}]_{\times}^T) \\ &\stackrel{(2.6)}{=} -2\boldsymbol{\xi}^T \mathbf{Z}^T (\mathbf{I} \otimes \text{mat}(\mathbf{A}\mathbf{r} + \mathbf{b}) \mathbf{R}^T) \mathbf{Z} \boldsymbol{\xi} \\ &= \boldsymbol{\xi}^T \text{Hess}_1(f) \boldsymbol{\xi} , \end{aligned} \quad (9.24)$$

$$\begin{aligned} 2\text{vec}([\boldsymbol{\xi}]_{\times} \mathbf{R})^T \mathbf{A}\text{vec}([\boldsymbol{\xi}]_{\times} \mathbf{R}) &= 2\boldsymbol{\xi}^T \mathbf{Z}^T (\mathbf{R} \otimes \mathbf{I}) \mathbf{A} (\mathbf{R} \otimes \mathbf{I})^T \mathbf{Z} \boldsymbol{\xi} \\ &= \boldsymbol{\xi}^T \text{Hess}_2(f) \boldsymbol{\xi} . \end{aligned} \quad (9.25)$$

Finally, the Hessian of the locally parameterized objective function can be obtained:

$$\text{Hess}(f) = \text{Hess}_1(f) + \text{Hess}_2(f) \quad (9.26)$$

$$= 2\mathbf{Z}^T ((\mathbf{R} \otimes \mathbf{I}) \mathbf{A} (\mathbf{R} \otimes \mathbf{I})^T - \mathbf{I} \otimes \text{mat}(\mathbf{A}\mathbf{r} + \mathbf{b}) \mathbf{R}^T) \mathbf{Z} . \quad (9.27)$$

9.1.3 Proofs

Invertibility of \mathbf{A}_{tt}

Calculating the translation vector from the rotation in Sec. 5.2.5 requires the matrix \mathbf{A}_{tt} to be invertible. Here, it is shown that \mathbf{A}_{tt} is positive definite in most cases and thus invertible. It needs to be shown:

$$\mathbf{x}^T \mathbf{A}_{tt} \mathbf{x} > 0 \implies \mathbf{A}_{tt} \text{ is invertible.} \quad (9.28)$$

With $\|\mathbf{d}_i\| = 1, w_{ik} > 0$ and $\forall \mathbf{x} \in \mathbb{R}^3$ with $\|\mathbf{x}\| > 0$ it follows:

$$\begin{aligned} \mathbf{x}^T \mathbf{A}_{tt} \mathbf{x} &= \mathbf{x}^T \sum_i w_{ik} [\mathbf{d}_i]_{\times} [\mathbf{d}_i]_{\times}^T \mathbf{x} = \sum_i w_{ik} \mathbf{x}^T [\mathbf{d}_i]_{\times} [\mathbf{d}_i]_{\times}^T \mathbf{x} \\ &= \sum_i w_{ik} ([\mathbf{d}_i]_{\times}^T \mathbf{x})^T [\mathbf{d}_i]_{\times}^T \mathbf{x} = \sum_i w_{ik} \|[\mathbf{d}_i]_{\times}^T \mathbf{x}\|^2 \\ &= \sum_i w_{ik} \|\mathbf{x} \times \mathbf{d}_i\|^2 > 0. \end{aligned}$$

This is always true, except for the degenerate case of parallel rays, *e.g.*, orthographic projection, telecentric optics. Then $\mathbf{x} = s\mathbf{d}_i, \forall i$ with some arbitrary scalar s , results in $\mathbf{x}^T \mathbf{A}_{tt} \mathbf{x} = 0$. In this case, there is an ambiguity in the translation term, because it is not possible to estimate the distance between the calibration pattern and a camera with orthographic projection:

$$\mathbf{t} = \mathbf{t}_0 + s\mathbf{d}_0. \quad (9.29)$$

Convergence of AM-Calibration

Following the research in the field of alternating minimization [65, 139], the following proofs that the proposed alternating minimization technique for camera calibration is convergent. Thus

$$f(\mathcal{P}^{(n+1)}, \mathcal{L}^{(n+1)}) < f(\mathcal{P}^{(n)}, \mathcal{L}^{(n)}) \quad (9.30)$$

needs to be shown, where $\mathcal{L}^{(n)}$ is the set of ray parameters and $\mathcal{P}^{(n)} = [\mathcal{R}^{(n)}, \mathcal{T}^{(n)}]$ the set of pose parameters, consisting of rotations and translations.

Define the operators $\mathcal{S}_{\mathbf{L}}$ and $\mathcal{S}_{\mathbf{P}}$, as solution to the ray subproblem of Sec. 5.2.4 and as solution to the pose subproblem of Sec. 5.2.5, respectively:

$$\mathcal{S}_{\mathbf{L}} \{f(\mathcal{P}^{(n)}, \mathcal{L}^{(n)})\} = f(\mathcal{P}^{(n)}, \mathcal{L}^{(n+1)}) , \quad (9.31)$$

$$\mathcal{S}_{\mathbf{P}} \{f(\mathcal{P}^{(n)}, \mathcal{L}^{(n+1)})\} = f(\mathcal{P}^{(n+1)}, \mathcal{L}^{(n+1)}) . \quad (9.32)$$

Because the optimization of camera rays delivers an optimal solution to its subproblem, we cannot get an increase in the objective function:

$$\mathcal{S}_{\mathbf{L}} \{f(\mathcal{P}^{(n)}, \mathcal{L}^{(n)})\} \leq f(\mathcal{P}^{(n)}, \mathcal{L}^{(n)}) . \quad (9.33)$$

Furthermore, if the Newton descend algorithm for pose estimation is initialized with the previous pose, we always get a descend in the objective function value:

$$\mathcal{S}_{\mathbf{P}} \{f(\mathcal{P}^{(n)}, \mathcal{L}^{(n+1)})\} < f(\mathcal{P}^{(n)}, \mathcal{L}^{(n+1)}) . \quad (9.34)$$

In conclusion, it follows:

$$\begin{aligned} f(\mathcal{P}^{(n+1)}, \mathcal{L}^{(n+1)}) &= \mathcal{S}_{\mathbf{P}} \{f(\mathcal{P}^{(n)}, \mathcal{L}^{(n+1)})\} \\ &< f(\mathcal{P}^{(n)}, \mathcal{L}^{(n+1)}) \\ &= \mathcal{S}_{\mathbf{L}} \{f(\mathcal{P}^{(n)}, \mathcal{L}^{(n)})\} \\ &\leq f(\mathcal{P}^{(n)}, \mathcal{L}^{(n)}) , \end{aligned} \quad (9.35)$$

$$\Rightarrow f(\mathcal{P}^{(n+1)}, \mathcal{L}^{(n+1)}) < f(\mathcal{P}^{(n)}, \mathcal{L}^{(n)}) \text{ q.e.d.} \quad (9.36)$$

Bibliography

- [1] **P.-A. Absil, R. Mahony, and R. Sepulchre.** *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.
- [2] **Y. Adato, Y. Vasilyev, T. Zickler, and O. Ben-Shahar.** *Shape from specular flow*. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.11 (2010), pp. 2054–2070.
- [3] **E. H. Adelson and J. Y. A. Wang.** *Single lens stereo with a plenoptic camera*. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14.2 (1992), pp. 99–106.
- [4] **M. Alain and A. Smolic.** *Light Field Denoising by Sparse 5D Transform Domain Collaborative Filtering*. In: *IEEE International Workshop on Multimedia Signal Processing (MMSP 2017)*. 2017.
- [5] **A. Albarelli, L. Cosmo, F. Bergamasco, and A. Torsello.** *High-Coverage 3D Scanning through Online Structured Light Calibration*. In: *2014 22nd International Conference on Pattern Recognition*. IEEE, 2014, pp. 4080–4085.
- [6] **S. Allgeier, U. Gengenbach, B. Köhler, K.-M. Reichert, and V. Hagenmeyer.** *Reproducibility of two calibration procedures for phase-measuring deflectometry*. In: *Interferometry XX*. Vol. 11490. SPIE, 2020, pp. 82–95.
- [7] **A. Alperovich, O. Johannsen, and B. Goldluecke.** *Intrinsic Light Field Decomposition and Disparity Estimation with Deep Encoder-Decoder Network*. In: *2018 26th European Signal Processing Conference (EUSIPCO)*. 2018, pp. 2165–2169.
- [8] **D. Antensteiner, S. Štolc, and T. Pock.** *A Review of Depth and Normal Fusion Algorithms*. In: *Sensors (Basel, Switzerland)* 18.2 (2018).
- [9] **J. Balzer.** *Regularisierung des Deflektometrieproblems - Grundlagen und Anwendung*. PhD thesis. 2008.

- [10] **J. Balzer, S. Hofer, and J. Beyerer.** *Multiview specular stereo reconstruction of large mirror surfaces.* In: *CVPR 2011.* 2011, pp. 2537–2544.
- [11] **J. Balzer and S. Werling.** *Principles of Shape from Specular Reflection.* In: *Measurement* 43.10 (2010), pp. 1305–1317.
- [12] **A. Bartoli and P. Sturm.** *The 3D Line Motion Matrix and Alignment of Line Reconstructions.* In: *International Journal of Computer Vision* 57.3 (2004), pp. 159–178.
- [13] **J. Bartsch, M. Kalms, and R. B. Bergmann.** *Improving the calibration of phase measuring deflectometry by a polynomial representation of the display shape.* In: *Journal of the European Optical Society: Rapid Publications* 15.1 (2019), p. 20.
- [14] **F. Baselice, D. Chirico, G. Ferraioli, and G. Schirinzi.** *Probabilistic data association Kalman filter for multi-channel phase unwrapping.* In: *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS).* 2015, pp. 4260–4263.
- [15] **F. Bergamasco, A. Albarelli, E. Rodola, and A. Torsello.** *Can a Fully Unconstrained Imaging Model Be Applied Effectively to Central Cameras?* In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013.* Piscataway, NJ: IEEE, 2013, pp. 1391–1398.
- [16] **J. Beyerer, F. Puente León, and C. Frese.** *Machine vision: Automated visual inspection: Theory, practice and applications.* Springer Verlag, 2016. 798 pp.
- [17] **D. N. Bhat and S. K. Nayar.** *Stereo and Specular Reflection.* In: *International Journal of Computer Vision* 26.2 (1998), pp. 91–106.
- [18] **J. Bigun.** *Optimal Orientation Detection of Linear Symmetry.* Tech. rep. 828. Linköping University, The Institute of Technology, 1987, pp. 433–438.
- [19] **T. E. Bishop, S. Zanetti, and P. Favaro.** *Light field superresolution.* In: *IEEE International Conference on Computational Photography (ICCP), 2009.* Piscataway, NJ: IEEE, 2009, pp. 1–9.

- [20] **Y. Bok, H.-G. Jeon, and I. S. Kweon.** *Geometric Calibration of Micro-Lens-Based Light Field Cameras Using Line Features.* In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.2 (2017), pp. 287–300.
- [21] **T. Bonfort and P. Sturm.** *Voxel Carving for Specular Surfaces.* In: *9th IEEE International Conference on Computer Vision (ICCV '03).* Vol. 1. Nice, France: IEEE Computer Society, 2003, pp. 691–696.
- [22] **T. Bothe, W. Li, C. von Kopylow, and W. P. O. Juptner.** *High-resolution 3D shape measurement on specular surfaces by fringe reflection.* In: *Optical Metrology in Production Engineering.* Vol. 5457. SPIE, 2004, pp. 411–422.
- [23] **T. Bothe, W. Li, M. Schulte, C. v. Kopylow, R. B. Bergmann, and W. P. O. Jüptner.** *Vision ray calibration for the quantitative geometric description of general imaging and projection optics in metrology.* In: *Applied Optics* 49.30 (2010), pp. 5851–5860.
- [24] **N. Boumal.** *Optimization and estimation on manifolds.* PhD thesis. Catholic University of Louvain, Louvain-la-Neuve, Belgium, 2014.
- [25] **S. P. Boyd and L. Vandenberghe.** *Convex optimization.* 18. print. Cambridge: Cambridge Univ. Press, 2015.
- [26] **G. Bradski.** *The OpenCV Library.* In: *Dr. Dobb's Journal of Software Tools* (2000).
- [27] **K. Bredies and M. Holler.** *A TGV-Based Framework for Variational Image Decompression, Zooming, and Reconstruction. Part II: Numerics.* In: *SIAM Journal on Imaging Sciences* 8.4 (2015), pp. 2851–2886.
- [28] **K. Bredies, K. Kunisch, and T. Pock.** *Total Generalized Variation.* In: *SIAM Journal on Imaging Sciences* 3.3 (2010), pp. 492–526.
- [29] **M. Breitbarth, P. Kühmstedt, and G. Notni.** *Calibration of a combined system with phase measuring deflectometry and fringe projection.* In: *SPIE Proceedings.* SPIE, 2009, p. 738909.
- [30] **R. P. Brent.** *Algorithms for minimization without derivatives.* Courier Corporation, 2013.

- [31] **O. N. Bria, J. G. Proakis, and D. G. Manolakis.** *Digital signal processing. Principles algorithms and applications.* In: *Journal of Computer Science and Technology* 1.01 (1999), 1 p.
- [32] **G. F. Carballo and P. W. Fieguth.** *Probabilistic cost functions for network flow phase unwrapping.* In: *IEEE Transactions on Geoscience and Remote Sensing* 38.5 (2000), pp. 2192–2201.
- [33] **A. Chambolle.** *An Algorithm for Total Variation Minimization and Applications.* In: *Journal of Mathematical Imaging and Vision* 20.1 (2004), pp. 89–97.
- [34] **A. Chambolle and T. Pock.** *A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging.* In: *Journal of Mathematical Imaging and Vision* 40.1 (2011), pp. 120–145.
- [35] **J. Y. Chang, K. M. Lee, and S. U. Lee.** *Multiview normal field integration using level set methods.* In: *IEEE Conference on Computer Vision and Pattern Recognition, 2007.* Los Alamitos, Calif.: IEEE Computer Society, 2007, pp. 1–8.
- [36] **C. Chen, J. Yu, N. Gao, and Z. Zhang.** *High accuracy 3D calibration method of phase calculation-based fringe projection system by using LCD screen considering refraction error.* In: *Optics and Lasers in Engineering* 126 (2020), p. 105870.
- [37] **Y.-Y. Cheng and J. C. Wyant.** *Two-wavelength phase shifting interferometry.* In: *Appl. Opt.* 23.24 (1984), pp. 4539–4543.
- [38] **D. Cho, M. Lee, S. Kim, and Y.-W. Tai.** *Modeling the Calibration Pipeline of the Lytro Camera for High Quality Light-Field Image Reconstruction.* In: *2013 IEEE International Conference on Computer Vision.* IEEE, 2013, pp. 3280–3287.
- [39] **T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein.** *Introduction to Algorithms, Third Edition.* 3rd. The MIT Press, 2009.
- [40] **M. Costantini, A. Farina, and F. Zirilli.** *A fast phase unwrapping algorithm for SAR interferometry.* In: *IEEE Transactions on Geoscience and Remote Sensing* 37.1 (1999), pp. 452–460.

- [41] **A. Criminisi, S. B. Kang, R. Swaminathan, R. Szeliski, and P. Anandan.** *Extracting layers and analyzing their specular properties using epipolar-plane-image analysis.* In: *Computer Vision and Image Understanding* 97.1 (2005), pp. 51–85.
- [42] **R. Dändliker, R. Thalmann, and D. Prongué.** *Two-wavelength laser interferometry using superheterodyne detection.* In: *Opt. Lett.* 13.5 (1988), pp. 339–341.
- [43] **D. G. Dansereau, A. Eriksson, and J. Leitner.** *Richardson-Lucy Deblurring for Moving Light Field Cameras.* In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops.* 2017.
- [44] **D. G. Dansereau, O. Pizarro, and S. B. Williams.** *Decoding, Calibration and Rectification for Lenselet-Based Plenoptic Cameras.* In: *2013 IEEE Conference on Computer Vision and Pattern Recognition.* 2013, pp. 1027–1034.
- [45] **Y. Ding, J. Xi, Y. Yu, and J. Chicharo.** *Recovering the absolute phase maps of two fringe patterns with selected frequencies.* In: *Opt. Lett.* 36.13 (2011), pp. 2518–2520.
- [46] **Y. Ding, J. Xi, Y. Yu, and F. Deng.** *Absolute phase recovery of three fringe patterns with selected spatial frequencies.* In: *Optics and Lasers in Engineering* 70 (2015), pp. 18–25.
- [47] **D. Droschel, D. Holz, and S. Behnke.** *Multi-frequency Phase Unwrapping for Time-of-Flight cameras.* In: *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems.* 2010, pp. 1463–1469.
- [48] **D. Droschel, D. Holz, and S. Behnke.** *Probabilistic Phase Unwrapping for Time-of-Flight Cameras.* In: *ISR 2010 (41st International Symposium on Robotics) and ROBOTIK 2010 (6th German Conference on Robotics).* 2010, pp. 1–7.
- [49] **D. M. Endres and J. E. Schindelin.** *A new metric for probability distributions.* In: *IEEE Transactions on Information Theory* 49.7 (2003), pp. 1858–1860.
- [50] **S. Ettl, J. Kaminski, M. C. Knauer, and G. Häusler.** *Shape reconstruction from gradient data.* In: *Applied Optics* 47.12 (2008), p. 2091.

- [51] **C. Faber, E. Olesch, R. Krobot, and G. Häusler.** *Deflectometry challenges interferometry: The competition gets tougher!* In: International Society for Optics and Photonics, 2012.
- [52] **J. M. Facil, B. Ummenhofer, H. Zhou, L. Montesano, T. Brox, and J. Civera.** *CAM-Convs: Camera-Aware Multi-Scale Convolutions for Single-View Depth.* In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.
- [53] **K. Falaggis, D. P. Towers, and C. E. Towers.** *Multiwavelength interferometry: extended range metrology.* In: *Opt. Lett.* 34.7 (2009), pp. 950–952.
- [54] **K. Falaggis, D. P. Towers, and C. E. Towers.** *Method of excess fractions with application to absolute distance metrology: theoretical analysis.* In: *Appl. Opt.* 50.28 (2011), pp. 5484–5498.
- [55] **K. Falaggis, D. P. Towers, and C. E. Towers.** *Method of excess fractions with application to absolute distance metrology: Wavelength selection and the effects of common error sources.* In: *Appl. Opt.* 51.27 (2012), pp. 6471–6479.
- [56] **K. Falaggis, D. P. Towers, and C. E. Towers.** *Method of excess fractions with application to absolute distance metrology: analytical solution.* In: *Appl. Opt.* 52.23 (2013), pp. 5758–5765.
- [57] **K. Falaggis, D. P. Towers, and C. E. Towers.** *Algebraic solution for phase unwrapping problems in multiwavelength interferometry.* In: *Appl. Opt.* 53.17 (2014), pp. 3737–3747.
- [58] **V. Farber, Y. Oiknine, I. August, and A. Stern.** *Spectral light fields for improved three-dimensional profilometry.* In: *Optical Engineering* 57.6 (2018), pp. 1–8.
- [59] **M. Fischer, M. Petz, and R. Tutsch.** *Model-based noise prediction for fringe projection systems - A tool for the statistical analysis of evaluation algorithms.* In: *tm - Technisches Messen* 84.2 (2017), pp. 111–122.
- [60] **T. Georgiev and A. Lumsdaine.** *Depth of Field in Plenoptic Cameras.* In: *Eurographics 2009 - Short Papers*. The Eurographics Association, 2009.
- [61] **T. Georgiev and A. Lumsdaine.** *The multifocus plenoptic camera.* In: International Society for Optics and Photonics, 2012.

- [62] **T. Goldstein, B. O'Donoghue, S. Setzer, and R. Baraniuk.** *Fast Alternating Direction Optimization Methods*. In: *SIAM Journal on Imaging Sciences* 7.3 (2014), pp. 1588–1623.
- [63] **G. H. Golub and C. F. van Loan.** *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, 2013.
- [64] **S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen.** *The lumigraph*. In: ACM, 1996.
- [65] **L. Grippo and M. Sciandrone.** *On the convergence of the block non-linear Gauss–Seidel method under convex constraints*. In: *Operations Research Letters* 26.3 (2000), pp. 127–136.
- [66] **M. D. Grossberg and S. K. Nayar.** *A general imaging model and a method for finding its parameters*. In: *Proceedings / Eighth IEEE International Conference on Computer Vision*. Los Alamitos, Calif.: IEEE Computer Society, 2001, pp. 108–115.
- [67] **M. D. Grossberg and S. K. Nayar.** *The Raxel Imaging Model and Ray-Based Calibration*. In: *International Journal of Computer Vision* 61.2 (2005), pp. 119–137.
- [68] **X. Guo, J. Tang, J. Li, C. Shen, and J. Liu.** *Attitude measurement based on imaging ray tracking model and orthographic projection with iteration algorithm*. In: *ISA Transactions* (2019).
- [69] **V.I. Gushov and Yu.N. Solodkin.** *Automatic processing of fringe patterns in integer interferometers*. In: *Optics and Lasers in Engineering* 14.4 (1991), pp. 311–324.
- [70] **W. Ha and R. F. Barber.** *Alternating minimization and alternating descent over nonconvex sets*. In: *arXiv preprint* (2017).
- [71] **K. Han, K.-Y. K. Wong, D. Schnieders, and M. Liu.** *Mirror Surface Reconstruction under an Uncalibrated Camera*. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2016, pp. 1772–1780.
- [72] **R. Hartley and A. Zisserman.** *Multiple View Geometry in Computer Vision*. 2nd ed. Cambridge University Press, 2004.
- [73] **R. Hartley, J. Trumpf, Y. Dai, and H. Li.** *Rotation Averaging*. In: *International Journal of Computer Vision* 103.3 (2013), pp. 267–305.

- [74] **G. Häusler, C. Richter, K.-H. Leitz, and M. C. Knauer.** *Microdeflectometry—a novel tool to acquire three-dimensional microtopography with nanometer height resolution.* In: *Opt. Lett.* 33.4 (2008), pp. 396–398.
- [75] **G. Häusler, M. C. Knauer, C. Faber, C. Richter, S. Peterhänsel, C. Kranitzky, and K. Veit.** *Deflectometry Challenges Interferometry: 3D-Metrology from Nanometer to Meter.* In: *Advances in Imaging.* Optica Publishing Group, 2009, p. DMC4.
- [76] **S. Heber and T. Pock.** *Convolutional Networks for Shape from Light Field.* In: *29th IEEE Conference on Computer Vision and Pattern Recognition.* Piscataway, NJ: IEEE, 2016, pp. 3746–3754.
- [77] **S. Heber, R. Ranftl, and T. Pock.** *Variational Shape from Light Field.* In: *Energy Minimization Methods in Computer Vision and Pattern Recognition.* Vol. 8081. Lecture Notes in Computer Science / Image Processing, Computer Vision, Pattern Recognition, and Graphics. Berlin/Heidelberg: Springer Berlin Heidelberg, 2013, pp. 66–79.
- [78] **S. Heber, W. Yu, and T. Pock.** *Neural EPI-Volume Networks for Shape From Light Field.* In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2017, pp. 2252–2260.
- [79] **J. Heikkila and O. Silven.** *A four-step camera calibration procedure with implicit image correction.* In: *Proceedings / 1997 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 17 - 19, 1997, San Juan, Puerto Rico.* Los Alamitos, Calif.: IEEE Computer Society, 1997, pp. 1106–1112.
- [80] **C. Heinze, S. Spyropoulos, S. Hussmann, and C. Perwass.** *Automated robust metric calibration of multi-focus plenoptic cameras.* In: *IEEE,* 2015.
- [81] **C. Hertzberg, R. Wagner, U. Frese, and L. Schröder.** *Integrating generic sensor fusion algorithms with sound state representations through encapsulation of manifolds.* In: *Information Fusion* 14.1 (2013), pp. 57–77.
- [82] **S. Höfer.** *Untersuchung diffus spiegelnder Oberflächen mittels Infrarotdeflektometrie.* German. PhD thesis. Karlsruher Institut für Technologie (KIT), 2017. 224 pp.

-
- [83] **S. Höfer, S. Werling, and J. Beyerer.** *Thermal pattern generation for infrared deflectometry.* In: *AMA Conferences 2013 - Nürnberg Exhibition Centre, May 14-16, 2013 - SENSOR, OPTO and IRS².* AMA, 2013, pp. 785–790.
- [84] **J. W. Horbach and T. Dang.** *3D reconstruction of specular surfaces using a calibrated projector–camera setup.* In: *Machine Vision and Applications* 21.3 (2010), pp. 331–340.
- [85] **L. Huang, J. Xue, B. Gao, C. McPherson, J. Beverage, and M. Idir.** *Modal phase measuring deflectometry.* In: *Optics Express* 24.21 (2016), pp. 24649–24664.
- [86] **L. Huang, M. Idir, C. Zuo, and A. Asundi.** *Review of phase measuring deflectometry.* In: *Optics and Lasers in Engineering* 107 (2018), pp. 247–257.
- [87] **R. Huang, P. Su, J. H. Burge, and M. Idir.** *X-ray mirror metrology using SCOTS/deflectometry.* In: *Advances in X-Ray/EUV Optics and Components VIII.* Vol. 8848. SPIE, 2013, pp. 110–115.
- [88] **J. M. Huntley and H. Saldner.** *Temporal phase-unwrapping algorithm for automated interferogram analysis.* In: *Appl. Opt.* 32.17 (1993), pp. 3047–3052.
- [89] **J. M. Huntley and H. O. Saldner.** *Error-reduction methods for shape measurement by temporal phase unwrapping.* In: *Journal of the Optical Society of America A* 14.12 (1997), p. 3188.
- [90] **J. M. Huntley and H. O. Saldner.** *Shape measurement by temporal phase unwrapping: Comparison of unwrapping algorithms.* In: *Measurement Science and Technology* 8.9 (1997), p. 986.
- [91] **Y. Ideguchi, Y. Uranishi, S. Yoshimoto, Y. Kuroda, and O.oshiro.** *Light Field Convergency: Implicit Photometric Consistency on Transparent Surface.* In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops.* 2017.
- [92] **I. Ihrke, J. Restrepo, and L. Mignard-Debise.** *Principles of Light Field Imaging: Briefly revisiting 25 years of research.* In: *IEEE Signal Processing Magazine* 33.5 (2016), pp. 59–69.
- [93] **S. R. Jammalamadaka and A. SenGupta.** *Topics in Circular Statistics.* World Scientific, 2001.

- [94] **O. Johannsen, A. Sulc, and B. Goldluecke.** *On Linear Structure from Motion for Light Field Cameras.* In: *2015 IEEE International Conference on Computer Vision (ICCV).* 2015, pp. 720–728.
- [95] **O. Johannsen, A. Sulc, and B. Goldluecke.** *Variational Separation of Light Field Layers.* In: *Vision, Modelling and Visualization (VMV).* 2015.
- [96] **O. Johannsen, A. Sulc, and B. Goldluecke.** *What Sparse Light Field Coding Reveals about Scene Structure.* In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2016, pp. 3262–3270.
- [97] **O. Johannsen, C. Heinze, B. Goldluecke, and C. Perwass.** *On the Calibration of Focused Plenoptic Cameras.* In: *GCPR Workshop on Imaging New Modalities.* 2013.
- [98] **O. Johannsen et al.** *A Taxonomy and Evaluation of Dense Light Field Depth Estimation Algorithms.* In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops.* 2017.
- [99] **H. Jonquière, L. M. Mugnier, and R. Mercier-Ythier.** *Study of linear phase shift algorithms and application to deflectometry.* In: *Optics and Lasers in Engineering* 143 (2021), p. 106640.
- [100] **S. Kammel and J. Horbach.** *Topography reconstruction of specular surfaces.* In: *Videometrics VIII.* Vol. 5665. SPIE, 2005, pp. 59–66.
- [101] **K. Kanatani.** *Analysis of 3-D rotation fitting.* In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16.5 (1994), pp. 543–549.
- [102] **L. Kinell and M. Sjö Dahl.** *Robustness of reduced temporal phase unwrapping in the measurement of shape.* In: *Appl. Opt.* 40.14 (2001), pp. 2297–2303.
- [103] **M. C. Knauer, J. Kaminski, and G. Hausler.** *Phase measuring deflectometry: a new approach to measure specular free-form surfaces.* In: *Optical Metrology in Production Engineering.* Vol. 5457. International Society for Optics and Photonics. SPIE, 2004, pp. 366–376.

-
- [104] **L. Kneip and P. Furgale.** *OpenGV: A unified and generalized approach to real-time calibrated geometric vision.* In: *IEEE International Conference on Robotics and Automation (ICRA)*, 2014. Piscataway, NJ: IEEE, 2014, pp. 1–8.
- [105] **R. Koetter, B. J. Frey, N. Petrovic, and D. C. Munson.** *Unwrapping phase images by propagating probabilities across graphs.* In: *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*. Vol. 3. 2001, 1845–1848 vol.3.
- [106] **G. Kurz, I. Gilitschenski, and U. D. Hanebeck.** *Efficient evaluation of the probability density function of a wrapped normal distribution.* In: *2014 sensor data fusion: trends, solutions, applications (SDF 2014)*. Piscataway, NJ: IEEE, 2014, pp. 1–5.
- [107] **J. Lai, J. Li, C. He, and F. Liu.** *A robust and effective phase-shift fringe projection profilometry method for the extreme intensity.* In: *Optik - International Journal for Light and Electron Optics* 179 (2019), pp. 810–818.
- [108] **G.H. Lee, B. Li, M. Pollefeys, and F. Fraundorfer.** *Minimal solutions for the multi-camera pose estimation problem.* In: *The International Journal of Robotics Research* 34.7 (2015), pp. 837–848.
- [109] **V. Lepetit, F. Moreno-Noguer, and P. Fua.** *EPnP: An Accurate $O(n)$ Solution to the PnP Problem.* In: *International Journal of Computer Vision* 81.2 (2008), p. 155.
- [110] **M. Levoy and P. Hanrahan.** *Light field rendering.* In: *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. New York, NY: ACM, 1996, pp. 31–42.
- [111] **H. Li, Y. Hu, T. Tao, S. Feng, M. Zhang, Y. Zhang, and C. Zuo.** *Optimal wavelength selection strategy in temporal phase unwrapping with projection distance minimization.* In: *Applied Optics* 57.10 (2018), pp. 2352–2360.
- [112] **J. Li, L. G. Hassebrook, and C. Guan.** *Optimized two-frequency phase-measuring-profilometry light-sensor temporal-noise sensitivity.* In: *Journal of the Optical Society of America A* 20.1 (2003), pp. 106–115.

- [113] **W. Li, M. Sandner, A. Gesierich, and J. Burke.** *Absolute optical surface measurement with deflectometry.* In: *Interferometry XVI: Applications*. Vol. 8494. International Society for Optics and Photonics. SPIE, 2012, 84940G.
- [114] **H. Liang, A. Zimmermann, R. Kickingeder, and C. Faber.** *A new method for solving the height problem in deflectometry.* In: International Society for Optics and Photonics, 2019, 111020P.
- [115] **E. Lilienblum and B. Michaelis.** *Optical 3D Surface Reconstruction by a Multi-Period Phase Shift Method.* In: *Journal of Computers* 2.2 (2007).
- [116] **G. Lippmann.** *Épreuves réversibles donnant la sensation du relief.* In: *J. Phys. Theor. Appl.* 7.1 (1908), pp. 821–825.
- [117] **K. Liu, Y. Wang, D. L. Lau, Q. Hao, and L. G. Hassebrook.** *Gamma model and its analysis for phase measuring profilometry.* In: *Journal of the Optical Society of America. A, Optics, image science, and vision* 27.3 (2010), pp. 553–562.
- [118] **M. Liu, R. Hartley, and M. Salzmann.** *Mirror Surface Reconstruction from a Single Image.* In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37.4 (2015), pp. 760–773.
- [119] **X. Liu, Z. Cai, X. Peng, and B. Z. Gao.** *Fringe Projection Profilometry Performed with a Light Field Camera.* In: *Imaging and Applied Optics 2018 (3D, AO, AIO, COSI, DH, IS, LACSEA, LS&C, MATH, pcAOP)*. Optica Publishing Group, 2018, DTh4B.5.
- [120] **Y. Liu, S. Huang, Z. Zhang, N. Gao, F. Gao, and X. Jiang.** *Full-field 3D shape measurement of discontinuous specular objects by direct phase measuring deflectometry.* In: *Scientific Reports* 7.1 (2017), p. 10293.
- [121] **F. Lu, L. He, S. You, X. Chen, and Z. Hao.** *Identifying Surface BRDF from a single 4D Light Field Image via Deep Neural Network.* In: *IEEE Journal of Selected Topics in Signal Processing* (2017).
- [122] **A. Lumsdaine.** *Focused plenoptic camera and rendering.* In: *Journal of Electronic Imaging* 19.2 (2010), p. 021106.
- [123] **A. Lumsdaine and T. Georgiev.** *The focused plenoptic camera.* In: *2009 IEEE International Conference on Computational Photography (ICCP)*. 2009, pp. 1–8.

-
- [124] **H. Ma, H. Li, Z. Qian, S. Shi, and T. Mu.** *VommaNet: an End-to-End Network for Disparity Estimation from Reflective and Texture-less Light Field Images*. In: *arXiv preprint* (2018).
 - [125] **Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry.** *An Invitation to 3-D Vision*. Springer New York, 2004.
 - [126] **K. Maeno, H. Nagahara, A. Shimada, and R.-i. Taniguchi.** *Light Field Distortion Feature for Transparent Object Recognition*. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. Piscataway, NJ: IEEE, 2013, pp. 2786–2793.
 - [127] **D. Maestro-Watson, A. Izaguirre, and N. Arana-Arexolaleiba.** *LCD screen calibration for deflectometric systems considering a single layer refraction model*. In: *2017 IEEE International Workshop of Electronics, Control, Measurement, Signals and their Application to Mechatronics (ECMSM)*. IEEE, 2017, pp. 1–6.
 - [128] **A. Manakov, J. F. Restrepo, O. Klehm, R. Hegedüs, E. Eisemann, H.-P. Seidel, and I. Ihrke.** *A Reconfigurable Camera Add-on for High Dynamic Range, Multispectral, Polarization, and Light-Field Imaging*. In: *ACM Trans. Graph.* 32.4 (2013).
 - [129] **B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng.** *NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis*. In: *Computer Vision – ECCV 2020*. Cham: Springer International Publishing, 2020, pp. 405–421.
 - [130] **P. Miraldo and H. Araujo.** *Calibration of smooth camera models*. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.9 (2013), pp. 2091–2103.
 - [131] **P. Miraldo, H. Araujo, and J. Queiro.** *Point-based calibration using a parametric representation of the general imaging model*. In: *IEEE International Conference on Computer Vision (ICCV)*, 2011. Piscataway, NJ: IEEE, 2011, pp. 2304–2311.
 - [132] **P. Moon and D. E. Spencer.** *The photic field*. In: *Cambridge Mass: MIT Press* (1981).
 - [133] **F. Moreno-Noguer, V. Lepetit, and P. Fua.** *Accurate non-iterative $O(n)$ solution to the pnp problem*. In: *Computer vision, 2007. ICCV 2007. IEEE 11th international conference on.* 2007, pp. 1–8.

- [134] **Y. E. Nesterov.** *A method for solving the convex programming problem with convergence rate $O(1/k^2)$.* In: *Dokl. akad. nauk Sssr.* Vol. 269. 1983, pp. 543–547.
- [135] **R. Ng.** *Fourier Slice Photography.* In: *ACM Trans. Graph.* 24.3 (2005), pp. 735–744.
- [136] **R. Ng and P. Hanrahan.** *Digital correction of lens aberrations in light field photography.* In: *International Optical Design Conference 2006.* Vol. 6342. International Society for Optics and Photonics. SPIE, 2006, 63421E.
- [137] **R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan.** *Light field photography with a hand-held plenoptic camera.* In: *Computer Science Technical Report CSTR 2.11* (2005), pp. 1–11.
- [138] **Y. Ni, X. Wang, and L. Yin.** *Relative pose estimation for multiple cameras using Lie algebra optimization.* In: *Applied Optics* 58.11 (2019), pp. 2963–2972.
- [139] **U. Niesen, D. Shah, and G. Wornell.** *Adaptive Alternating Minimization Algorithms.* In: *IEEE International Symposium on Information Theory, 2007.* Piscataway, NJ: IEEE Service Center, 2007, pp. 1641–1645.
- [140] **J. Nocedal and S. J. Wright.** *Numerical optimization.* Second edition. Springer series in operation research and financial engineering. New York, NY: Springer, 2006.
- [141] **J. R. Nüss, M. Prinzler, M. Kalms, R. B. Bergmann, and J. Bartsch.** *Effects of non-ideal display properties in phase measuring deflectometry: A model-based investigation.* In: *Optical Micro- and Nanometrology VII.* SPIE, 2018, p. 35.
- [142] **E. Olesch, G. Häusler, A. Wörnlein, F. Stinzing, and C. van Eldik.** *Deflectometric measurement of large mirrors.* In: *Advanced Optical Technologies* 3.3 (2014), p. 8861.
- [143] **A. Pak.** *Stability of absolute depth reconstruction from deflectometric measurement data.* In: *Interferometry XVII: Techniques and Analysis.* Vol. 9203. International Society for Optics and Photonics. SPIE, 2014, 92030E.

-
- [144] **A. Pak.** *Measuring shape of a mirror with a moving camera.* In: *Modeling Aspects in Optical Metrology VI*. Vol. 10330. International Society for Optics and Photonics. SPIE, 2017, 103300O.
- [145] **D. S. Pankaj, R. R. Nidamanuri, and P. B. Prasad.** *3-D imaging techniques and review of products.* In: *Proceedings of International Conference on Innovations in Computer Science and Engineering*. 2013.
- [146] **N. Parikh and S. Boyd.** *Proximal Algorithms.* In: *Found. Trends Optim.* 1.3 (2014), pp. 127–239.
- [147] **X. Peng, Z. Yang, and H. Niu.** *Multi-resolution reconstruction of 3-D image with modified temporal unwrapping algorithm.* In: *Optics Communications* 224.1 (2003), pp. 35–44.
- [148] **C. Perwass and L. Wietzke.** *Single lens 3D-camera with extended depth-of-field.* In: International Society for Optics and Photonics, 2012.
- [149] **T. Petković, T. Pribanić, and M. Đonlić.** *Temporal phase unwrapping using orthographic projection.* In: *Optics and Lasers in Engineering* 90 (2017), pp. 34–47.
- [150] **M. Petz, H. Dierke, and R. Tutsch.** *Wellenlängenoptimierung bei Heterodyn-Phasenschiebverfahren.* In: *tm - Technisches Messen* 87.10 (2020), pp. 599–613.
- [151] **M. Petz and R. Ritter.** *Reflection grating method for 3D measurement of reflecting surfaces.* In: *Optical Measurement Systems for Industrial Inspection II: Applications in Production Engineering*. Vol. 4399. SPIE, 2001, pp. 35–41.
- [152] **M. Petz and R. Tutsch.** *Measurement of optically effective surfaces by imaging of gratings.* In: *Optical Measurement Systems for Industrial Inspection III*. Vol. 5144. SPIE, 2003, pp. 288–294.
- [153] **M. Petz and R. Tutsch.** *Rasterreflexions-Photogrammetrie zur Messung spiegelnder Oberflächen (Reflection Grating Photogrammetry for the Measurement of Specular Surfaces).* In: *tm - Technisches Messen* 71.7-8-2004 (2004), pp. 389–397.

- [154] **M. Petz and R. Tutsch.** *Reflection grating photogrammetry: a technique for absolute shape measurement of specular free-form surfaces.* In: *Optical Manufacturing and Testing VI*. Vol. 5869. SPIE, 2005, pp. 355–366.
- [155] **L. S. Ping, C. Y. Kit, L. T. Sze, and K. V. Chet.** *Noise reduction methods for terrain phase estimation of InSAR images.* In: *2016 IEEE 12th International Colloquium on Signal Processing Its Applications (CSPA)*. 2016, pp. 167–172.
- [156] **R. Pless.** *Using many cameras as one.* In: *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Los Alamitos, Calif: IEEE Computer Society, 2003, pp. II-587–93.
- [157] **T. Pock, L. Zebedin, and H. Bischof.** *TGV-Fusion.* In: *Rainbow of Computer Science: Dedicated to Hermann Maurer on the Occasion of His 70th Birthday*. Springer Berlin Heidelberg, 2011, pp. 245–258.
- [158] **C. Polhemus.** *Two-Wavelength Interferometry.* In: *Appl. Opt.* 12.9 (1973), pp. 2071–2074.
- [159] **R. Porras-Aguilar and K. Falaggis.** *Absolute phase recovery in structured light illumination systems: Sinusoidal vs. intensity discrete patterns.* In: *Optics and Lasers in Engineering* 84 (2016), pp. 111–119.
- [160] **T. Pribanić, H. Džapo, and J. Salvi.** *Efficient and Low-Cost 3D Structured Light System Based on a Modified Number-Theoretic Approach.* In: *EURASIP Journal on Advances in Signal Processing* 2010.1 (2009), p. 474389.
- [161] **T. Pribanić, S. Mrvoš, and J. Salvi.** *Efficient multiple phase shift patterns for dense 3D acquisition in structured light scanning.* In: *Image and Vision Computing* 28.8 (2010), pp. 1255–1266.
- [162] **Y. Quéau and J.-D. Durou.** *Edge-Preserving Integration of a Normal Field: Weighted Least-Squares, TV and L^1 Approaches.* In: *Scale Space and Variational Methods in Computer Vision*. Cham: Springer International Publishing, 2015, pp. 576–588.
- [163] **Y. Quéau, J.-D. Durou, and J.-F. Aujol.** *Normal Integration: A Survey.* In: *Journal of Mathematical Imaging and Vision* 60.4 (2018), pp. 576–593.

-
- [164] **Y. Quéau, J.-D. Durou, and J.-F. Aujol.** *Variational Methods for Normal Integration*. In: *Journal of Mathematical Imaging and Vision* 60.4 (2018), pp. 609–632.
- [165] **S. Ramalingam, P. Sturm, and S. K. Lodha.** *Towards Complete Generic Camera Calibration*. In: *CVPR 2005*. Los Alamitos, Calif: IEEE Computer Society, 2005, pp. 1093–1098.
- [166] **A. H. Ramirez-Andrade, R. Porras-Aguilar, and K. Falaggis.** *Numerical integration of slope data with application to deflectometry*. In: *Interferometry XX*. Vol. 11490. SPIE, 2020, pp. 41–48.
- [167] **C. Rathjen.** *Statistical properties of phase-shift algorithms*. In: *J. Opt. Soc. Am. A* 12.9 (1995), pp. 1997–2008.
- [168] **T. Reh, W. Li, J. Burke, and R. B. Bergmann.** *Improving the Generic Camera Calibration technique by an extended model of calibration display*. In: *Journal of the European Optical Society: Rapid Publications* 9 (2014).
- [169] **C. Reich, R. Ritter, and J. Thesing.** *White light heterodyne principle for 3D-measurement*. In: *Sensors, Sensor Systems, and Sensor Data Processing*. Vol. 3100. SPIE, 1997, pp. 236–244.
- [170] **C. Reich, R. Ritter, and J. Thesing.** *3-D shape measurement of complex objects by combining photogrammetry and fringe projection*. In: *Optical Engineering* 39.1 (2000), pp. 224–231.
- [171] **D. Rosebrock.** *Surface Model - An Uncertain Continuous Representation of the Generic Camera Model and its Calibration*. PhD thesis. Technischen Universität Carolo-Wilhelmina zu Braunschweig, 2016.
- [172] **S. Roth and M. J. Black.** *Specular Flow and the Recovery of Surface Structure*. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition workshops*. IEEE, 2006, pp. 1869–1876.
- [173] **J. Salvi, J. Pagès, and J. Batlle.** *Pattern codification strategies in structured light systems*. In: *Pattern Recognition* 37.4 (2004), pp. 827–849.

- [174] **A.C. Sanderson, L.E. Weiss, and S.K. Nayar.** *Structured high-light inspection of specular surfaces.* In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 10.1 (1988), pp. 44–55.
- [175] **G. Sansoni, M. Carocci, and R. Rodella.** *Three-dimensional vision based on a combination of gray-code and phase-shift light projection: analysis and compensation of the systematic errors.* In: *Appl. Opt.* 38.31 (1999), pp. 6565–6573.
- [176] **M. Sarkis and K. Diepold.** *Camera-pose estimation via projective Newton optimization on the manifold.* In: *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society* 21.4 (2012), pp. 1729–1741.
- [177] **S. Savarese, M. Chen, and P. Perona.** *Local Shape from Mirror Reflections.* In: *International Journal of Computer Vision* 64.1 (2005), pp. 31–67.
- [178] **M. Schambach.** *Reconstruction from Spatio-Spectrally Coded Multispectral Light Fields.* PhD thesis. Karlsruher Institut für Technologie (KIT), 2022.
- [179] **M. Schambach and M. Heizmann.** *A Multispectral Light Field Dataset and Framework for Light Field Deep Learning.* In: *IEEE Access* 8 (2020), pp. 193492–193502.
- [180] **M. Schambach and F. Puente León.** *Microlens Array Grid Estimation, Light Field Decoding, and Calibration.* In: *IEEE Transactions on Computational Imaging* 6 (2020), pp. 591–603.
- [181] **M. Schambach, J. Shi, and M. Heizmann.** *Spectral Reconstruction and Disparity from Spatio-Spectrally Coded Light Fields via Multi-Task Deep Learning.* In: *Proceedings 2021 International Conference on 3D Vision: 3DV 2021 ; Virtual Conference, 1-3 December 2021.* Institute of Electrical and Electronics Engineers (IEEE), 2021, pp. 186–196.
- [182] **C. Schmalz, F. Forster, and E. Angelopoulos.** *Camera calibration: active versus passive targets.* In: *Optical Engineering* 50.11 (2011), p. 113601.
- [183] **R. S. Schone and O. Schwarz.** *Hybrid phase unwrapping algorithm extended by a minimum-cost-matching strategy.* In: *Speckle Metrology* 2003. Vol. 4933. SPIE, 2003, pp. 305–310.

-
- [184] **R. Seßner.** *Richtungscodierte Deflektometrie durch Telezentrie.* PhD thesis. Friedrich-Alexander-Universität Erlangen-Nürnberg, 2009.
- [185] **G. M. Shahzeb Khan and G. Bahadir K.** *Spatial and Angular Resolution Enhancement of Light Fields Using Convolutional Neural Networks.* In: *IEEE Transactions on Image Processing* 27.5 (2018), pp. 2146–2159.
- [186] **F. Shevlin.** *Analysis of orientation problems using Plucker lines.* In: *Proceedings.* Silver Spring, MD and Los Angeles, CA: IEEE Computer Society Press, 1984, pp. 685–689.
- [187] **C. Shin, H.-G. Jeon, Y. Yoon, I. So Kweon, and S. Joo Kim.** *EPINET: A Fully-Convolutional Neural Network Using Epipolar Geometry for Depth From Light Field Images.* In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2018.
- [188] **M. D. Shuster.** *Survey of attitude representations.* In: *Journal of the Astronautical Sciences* 41.4 (1993), pp. 439–517.
- [189] **J. Solà, J. Deray, and D. Atchuthan.** *A micro Lie theory for state estimation in robotics.* In: *arXiv preprint* (2018).
- [190] **J. Song, Y.-S. Ho, D. L. Lau, and K. Liu.** *Universal phase unwrapping for phase measuring profilometry using geometric analysis.* In: *Emerging Digital Micromirror Device Based Systems and Applications X.* Proceedings of SPIE. Bellingham, Washington, USA: SPIE, 2018, p. 9.
- [191] **P. Sturm and S. Ramalingam.** *A Generic Concept for Camera Calibration.* In: *Computer Vision - ECCV 2004.* Lecture Notes in Computer Science. Berlin and Heidelberg: Springer, 2004, pp. 1–13.
- [192] **T. Su, S. Wang, R. E. Parks, P. Su, and J. H. Burge.** *Measuring rough optical surfaces using scanning long-wave optical test system. 1. Principle and implementation.* In: *Appl. Opt.* 52.29 (2013), pp. 7117–7126.
- [193] **A. Sulc, A. Alperovich, N. Marniok, and B. Goldluecke.** *Reflection Separation in Light Fields based on Sparse Coding and Specular Flow.* In: *Vision, Modelling and Visualization (VMV).* 2016.

- [194] **Y. Surrel.** *Design of algorithms for phase measurements by the use of phase stepping.* In: *Appl. Opt.* 35.1 (1996), pp. 51–60.
- [195] **R. Swaminathan, S. B. Kang, R. Szeliski, A. Criminisi, and S. K. Nayar.** *On the Motion and Appearance of Specularities in Image Sequences.* In: *Computer Vision - ECCV 2002*. Vol. 2350. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2006, pp. 508–523.
- [196] **K. Takahashi, S. Nobuhara, and T. Matsuyama.** *A new mirror-based extrinsic camera calibration using an orthogonality constraint.* In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012*. Piscataway, NJ: IEEE, 2012, pp. 1051–1058.
- [197] **K. Takahashi, S. Nobuhara, and T. Matsuyama.** *Mirror-based camera pose estimation using an orthogonality constraint.* In: *IPSP Transactions on Computer Vision and Applications* 8 (2016), pp. 11–19.
- [198] **M. Takeda, Q. Gu, M. Kinoshita, H. Takai, and Y. Takahashi.** *Frequency-multiplex Fourier-transform profilometry: a single-shot three-dimensional shape measurement of objects with large height discontinuities and/or surface isolations.* In: *Appl. Opt.* 36.22 (1997), pp. 5347–5354.
- [199] **M. W. Tao, S. Hadap, J. Malik, and R. Ramamoorthi.** *Depth from Combining Defocus and Correspondence Using Light-Field Cameras.* In: *2013 IEEE International Conference on Computer Vision*. 2013, pp. 673–680.
- [200] **M. Tarini, H. P.A. Lensch, M. Goesele, and H.-P. Seidel.** *3D acquisition of mirroring objects using striped patterns.* In: *Graphical Models* 67.4 (2005), pp. 233–259.
- [201] **J. Tian, X. Peng, and X. Zhao.** *A generalized temporal phase unwrapping algorithm for three-dimensional profilometry.* In: *Optics and Lasers in Engineering* 46.4 (2008), pp. 336–342.
- [202] **C. E. Towers, D. P. Towers, and J. D.C. Jones.** *Time efficient Chinese remainder theorem algorithm for full-field fringe phase analysis in multi-wavelength interferometry.* In: *Opt. Express* 12.6 (2004), pp. 1136–1143.

-
- [203] **C. E. Towers, D. P. Towers, and Julian D. C. Jones.** *Optimum frequency selection in multifrequency interferometry.* In: *Optics Letters* 28.11 (2003), pp. 887–889.
- [204] **C.E. Towers, D.P. Towers, and J.D.C. Jones.** *Absolute fringe order calculation using optimised multi-frequency selection in full-field profilometry.* In: *Optics and Lasers in Engineering* 43.7 (2005), pp. 788–800.
- [205] **J. Unger, A. Wenger, T. Hawkins, A. Gardner, and P. Debevec.** *Capturing and Rendering With Incident Light Fields.* In: *Eurographics Workshop on Rendering.* The Eurographics Association, 2003.
- [206] **M. Unger, T. Pock, M. Werlberger, and H. Bischof.** *A Convex Approach for Variational Super-Resolution.* In: *Pattern Recognition.* Springer Berlin Heidelberg, 2010, pp. 313–322.
- [207] **W. v. d. Hodge and D. Pedoe.** *Methods of Algebraic Geometry.* Cambridge: Cambridge University Press, 1994.
- [208] **V. Vaish, B. Wilburn, N. Joshi, and M. Levoy.** *Using plane + parallax for calibrating dense camera arrays.* In: *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.* Vol. 1. 2004, p. I.
- [209] **S. van der Jeught and J. J.J. Dirckx.** *Real-time structured light profilometry: a review.* In: *Optics and Lasers in Engineering* 87 (2016), pp. 18–31.
- [210] **D. Varsamis, N. Karampetakis, and P. Mastorocostas.** *Transformations between Two-Variable Polynomial Bases with Applications.* In: *Applied Mathematics & Information Sciences* 10.4 (2016), pp. 1303–1311.
- [211] **A. Veeraraghavan, R. Raskar, A. Agrawal, A. Mohan, and J. Tumblin.** *Dappled Photography: Mask Enhanced Cameras for Heterodyned Light Fields and Coded Aperture Refocusing.* In: *ACM Trans. Graph.* 26.3 (2007), 69–es.
- [212] **A. Vianello, G. Manfredi, M. Diebold, and B. Jähne.** *3D reconstruction by a combined structure tensor and Hough transform light field approach.* In: *tm - Technisches Messen* 84.7-8 (2017), pp. 460–478.

- [213] **T.-C. Wang, J.-Y. Zhu, E. Hiroaki, M. Chandraker, A. A. Efros, and R. Ramamoorthi.** *A 4D light-field dataset and CNN architectures for material recognition.* In: *European Conference on Computer Vision.* 2016, pp. 121–138.
- [214] **Y. Wang and S. Zhang.** *Superfast multifrequency phase-shifting technique with optimal pulse width modulation.* In: *Opt. Express* 19.6 (2011), pp. 5149–5155.
- [215] **Y. Wang, G. Hou, Z. Sun, Z. Wang, and T. Tan.** *A simple and robust super resolution method for light field images.* In: *2016 IEEE International Conference on Image Processing.* Piscataway, NJ: IEEE, 2016, pp. 1459–1463.
- [216] **S. Wanner and B. Goldluecke.** *Globally consistent depth labeling of 4D light fields.* In: *2012 IEEE Conference on Computer Vision and Pattern Recognition.* 2012, pp. 41–48.
- [217] **S. Wanner and B. Goldluecke.** *Reconstructing Reflective and Transparent Surfaces from Epipolar Plane Images.* In: *Pattern Recognition.* Springer Berlin Heidelberg, 2013, pp. 1–10.
- [218] **S. Wanner and B. Goldluecke.** *Variational light field analysis for disparity estimation and super-resolution.* In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36.3 (2014), pp. 606–619.
- [219] **S. Wanner, C. Straehle, and B. Goldluecke.** *Globally Consistent Multi-label Assignment on the Ray Space of 4D Light Fields.* In: *2013 IEEE Conference on Computer Vision and Pattern Recognition.* 2013, pp. 1011–1018.
- [220] **S. Werling and J. Beyerer.** *Automatische Inspektion spiegelnder Oberflächen mittels inverser Muster (Inspection of Specular Surfaces with Inverse Patterns).* In: *tm - Technisches Messen* 74.4 (2007).
- [221] **S. Werling and J. Beyerer.** *Oberflächenrekonstruktion mittels Stereodeflektometrie.* In: *tm - Technisches Messen* 78.9 (2011), pp. 398–405.
- [222] **S. Werling, M. Mai, M. Heizmann, and J. Beyerer.** *Inspection of specular and partially specular surfaces.* In: *Metrology and Measurement Systems* 16.3 (2009), pp. 415–431.

- [223] **S. B. Werling.** *Deflektometrie zur automatischen Sichtprüfung und Rekonstruktion spiegelnder Oberflächen.* Schriftenreihe Automatische Sichtprüfung und Bildverarbeitung. Karlsruhe: KIT Scientific Publ, 2011.
- [224] **G. Wetzstein, R. Raskar, and W. Heidrich.** *Hand-held Schlieren Photography with Light Field probes.* In: 2011 IEEE International Conference on Computational Photography (ICCP). 2011, pp. 1–8.
- [225] **G. Wetzstein, D. Roodnick, W. Heidrich, and R. Raskar.** *Refractive shape from light field distortion.* In: IEEE International Conference on Computer Vision (ICCV), 2011. Piscataway, NJ: IEEE, 2011, pp. 1180–1186.
- [226] **Working group, EMVA.** *Emva Standard 1288 - Standard For Characterization Of Image Sensors And Cameras, Release A2.01.* 2007.
- [227] **Y.-L. Xiao, X. Su, and W. Chen.** *Flexible geometrical calibration for fringe-reflection 3D measurement.* In: Optics letters 37.4 (2012), pp. 620–622.
- [228] **Y.-L. Xiao, X. Su, W. Chen, and Y. Liu.** *Three-dimensional shape measurement of aspheric mirrors with fringe reflection photogrammetry.* In: Applied Optics 51.4 (2012), pp. 457–464.
- [229] **Y.-L. Xiao, S. Li, Q. Zhang, J. Zhong, X. Su, and Z. You.** *Optical fringe-reflection deflectometry with bundle adjustment.* In: Optics and Lasers in Engineering 105 (2018), pp. 132–140.
- [230] **Y. Xu, F. Gao, and X. Jiang.** *Enhancement of measurement accuracy of optical stereo deflectometry based on imaging model analysis.* In: Optics and Lasers in Engineering 111 (2018), pp. 1–7.
- [231] **Y. Xu, F. Gao, Z. Zhang, and X. Jiang.** *A holistic calibration method with iterative distortion compensation for stereo deflectometry.* In: Optics and Lasers in Engineering 106 (2018), pp. 111–118.
- [232] **J. Yang, H. Li, Y. Dai, and R. T. Tan.** *Robust Optical Flow Estimation of Double-Layer Images under Transparency or Reflection.* In: 29th IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2016, pp. 1410–1419.

- [233] **J. C. Yang, M. Everett, C. Buehler, and L. McMillan.** *A Real-Time Distributed Light Field Camera*. In: *Proceedings of the 13th Eurographics Workshop on Rendering*. EGRW '02. Goslar, DEU: Eurographics Association, 2002, pp. 77–86.
- [234] **H. W. F. Yeung, J. Hou, X. Chen, J. Chen, Z. Chen, and Y. Y. Chung.** *Light Field Spatial Super-Resolution Using Deep Efficient Spatial-Angular Separable Convolution*. In: *IEEE Transactions on Image Processing* 28.5 (2019), pp. 2319–2330.
- [235] **Y. Yoon, H.-G. Jeon, D. Yoo, J.-Y. Lee, and I. S. Kweon.** *Learning a Deep Convolutional Network for Light-Field Image Super-Resolution*. In: *2015 IEEE International Conference on Computer Vision workshops*. Piscataway, NJ: IEEE, 2015, pp. 57–65.
- [236] **A. Yu, S. Fridovich-Keil, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa.** *Plenoxels: Radiance Fields without Neural Networks*. In: *arXiv preprint* (2021).
- [237] **H.-M. Yue, Y.-X. Wu, Y.-P. Song, and Y. Liu.** *Solution to the slope-height ambiguity problem in phase measuring deflectometry based on a co-axial telecentric optical path*. In: *Measurement Science and Technology* 31.4 (2020), p. 045007.
- [238] **N. Zeller, F. Quint, and U. Stilla.** *From the Calibration of a Light-Field Camera to Direct Plenoptic Odometry*. In: *IEEE Journal of Selected Topics in Signal Processing* (2017), p. 1.
- [239] **N. Zeller, C. A. Noury, F. Quint, C. Teulière, U. Stilla, and M. Dhome.** *Metric Calibration of a Focused Plenoptic Camera Based on a 3D Calibration Target*. In: *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* III-3 (2016), pp. 449–456.
- [240] **Q. Zhang, X. Su, L. Xiang, and X. Sun.** *3-D shape measurement based on complementary Gray-code light*. In: *Optics and Lasers in Engineering* 50.4 (2012), pp. 574–579.
- [241] **S. Zhang.** *Comparative study on passive and active projector nonlinear gamma calibration*. In: *Appl. Opt.* 54.13 (2015), pp. 3834–3841.
- [242] **S. Zhang.** *Absolute phase retrieval methods for digital fringe projection profilometry: A review*. In: *Optics and Lasers in Engineering* 107 (2018), pp. 28–37.

- [243] **S. Zhang, X. Li, and S.-T. Yau.** *Multilevel quality-guided phase unwrapping algorithm for real-time three-dimensional shape reconstruction.* In: *Appl. Opt.* 46.1 (2007), pp. 50–57.
- [244] **S. Zhang, H. Sheng, C. Li, J. Zhang, and Z. Xiong.** *Robust depth estimation for light field via spinning parallelogram operator.* In: *Computer Vision and Image Understanding* 145 (2016), pp. 148–159.
- [245] **Y. Zhang, H. Lv, Y. Liu, H. Wang, X. Wang, Q. Huang, X. Xiang, and Q. Dai.** *Light-Field Depth Estimation via Epipolar Plane Image Analysis and Locally Linear Embedding.* In: *IEEE Transactions on Circuits and Systems for Video Technology* 27.4 (2017), pp. 739–747.
- [246] **Z. Zhang.** *A flexible new technique for camera calibration.* In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.11 (2000), pp. 1330–1334.
- [247] **P. Zhao, N. Gao, Z. Zhang, F. Gao, and X. Jiang.** *Performance analysis and evaluation of direct phase measuring deflectometry.* In: *Optics and Lasers in Engineering* 103 (2018), pp. 24–33.
- [248] **J. Zhong and M. Wang.** *Phase unwrapping by lookup table method: application to phase map with singular points.* In: *Optical Engineering* 38.12 (1999), pp. 2075–2080.
- [249] **J. Zhong and Y. Zhang.** *Absolute phase-measurement technique based on number theory in multifrequency grating projection profilometry.* In: *Appl. Opt.* 40.4 (2001), pp. 492–500.
- [250] **K. Zhou, M. Zaitsev, and S. Bao.** *Reliable two-dimensional phase unwrapping method using region growing and local linear estimation.* In: *Magnetic resonance in medicine* 62.4 (2009), pp. 1085–1090.
- [251] **P. Zhou, Y. Zhang, Y. Yu, W. Cai, and G. Zhou.** *3D reconstruction from structured light field by Fourier transformation profilometry.* In: *AOPC 2019: Optical Sensing and Imaging Technology*. Vol. 11338. SPIE, 2019, pp. 399–405.
- [252] **W. Ziwei, Y. Yang, L. Xiaoli, M. Yupei, H. Quanyao, Y. Yongkai, C. Zewei, T. Qijian, and P. Xiang.** *Light-Field-Assisted Phase Unwrapping of Fringe Projection Profilometry.* In: *IEEE Access* 9 (2021), pp. 49890–49900.

- [253] **C. Zuo, Q. Chen, G. Gu, S. Feng, F. Feng, R. Li, and G. Shen.** *High-speed three-dimensional shape measurement for dynamic scenes using bi-frequency tripolar pulse-width-modulation fringe projection.* In: *Optics and Lasers in Engineering* 51.8 (2013), pp. 953–960.
- [254] **C. Zuo, L. Huang, M. Zhang, Q. Chen, and A. Asundi.** *Temporal phase unwrapping algorithms for fringe projection profilometry: A comparative review.* In: *Optics and Lasers in Engineering* 85 (2016), pp. 84–103.
- [255] **C. Zuo, T. Tao, S. Feng, L. Huang, A. Asundi, and Q. Chen.** *Micro Fourier Transform Profilometry (μ FTP): 3D shape measurement at 10,000 frames per second.* In: *Optics and Lasers in Engineering* 102 (2018), pp. 70–91.

List of Publications

- [A1] David Uhlig and Michael Heizmann. *A Probabilistic Approach for Spatio-Temporal Phase Unwrapping in Multi-Frequency Phase-Shift Coding.* In: *IEEE Access* (2022), p. 21.
- [A2] David Uhlig and Michael Heizmann. *Temporale Phasenentfaltung mittels zirkulärer Statistik und lokalen Nachbarschaftsinformationen.* In: *DGaO-Proceedings* A25 (2021).
- [A3] David Uhlig and Michael Heizmann. *Model-independent light field reconstruction using a generic camera calibration.* In: *tm - Technisches Messen* 88.6 (2021), pp. 361–373.
- [A4] David Uhlig and Michael Heizmann. *Light Field Reconstruction using a Generic Imaging Model.* In: *Forum Bildverarbeitung 2020.* Ed.: T. Längle ; M. Heizmann. KIT Scientific Publishing, 2020, pp. 91–103.
- [A5] David Uhlig and Michael Heizmann. *A Calibration Method for the Generalized Imaging Model with Uncertain Calibration Target Coordinates.* In: *Proceedings of the Asian Conference on Computer Vision (ACCV).* 2020.

- [A6] David Uhlig and Michael Heizmann. *Lichtfeld-Tiefenschätzung für die Vermessung teilspiegelnder Oberflächen*. In: *tm - Technisches Messen* 87.s1 (2020), s9–s15.
- [A7] David Uhlig and Michael Heizmann. *Temporale Phasenentfaltung für Phasenmessende 3D-Messtechnik mittels zirkulärer Statistik*. In: *DGaO-Proceedings* A15 (2020).
- [A8] Thomas Nürnberg, Maximilian Schambach, David Uhlig, Michael Heizmann, and Fernando Puente León. *A simulation framework for the design and evaluation of computational cameras*. In: *Proceedings of SPIE* 11061 (2019), Article no: 1106102.
- [A9] David Uhlig and Michael Heizmann. *Multi-Stereo-Deflektometrie mit einer Lichtfeldkamera / Multi-stereo deflectometry with a light-field camera*. In: *tm - Technisches Messen* 85.s1 (2018), s59–s65.

List of Supervised Theses

- [B1] **F. Seiler**. *Analyse der Epipolarebenen auf teilspiegelnden Oberflächen*. IIIT, 2018.
- [B2] **X. He**. *Analyse eines Verfahrens zur Rekonstruktion spiegelnder Oberflächen*. IIIT, IAI, 2018.
- [B3] **A. Sahnoun**. *Lichtfeldtiefenschätzung auf teilspiegelnden Oberflächen*. IIIT, 2018.
- [B4] **M. S. Gssim**. *Modellfreie Kalibrierung einer Lichtfeldkamera*. IIIT, 2018.
- [B5] **K. Zhou**. *Optischer Fluss im Lichtfeld und Tiefenschätzung bei Teilspiegelung oder Transparenz*. IIIT, 2018.
- [B6] **M. Reichardt**. *Lichtfeld-Tiefenberechnung von teilspiegelnden Oberflächen mit künstlichen neuronalen Netzen*. IIIT, 2019.
- [B7] **A. Gutermann**. *Analyse von gekrümmten Spiegeloberflächen mittels Lichtfeldtiefenschätzung*. IIIT, 2019.
- [B8] **K. Shen**. *Regularization and Data Fusion in Reconstruction of Reflective Surfaces*. IIIT, 2019.

- [B9] **M. Hügel.** *Deflektometrische Rekonstruktion spiegelnder Oberflächen mithilfe eines generalisierten Modells einer Lichtfeldkamera.* IIIT, 2020.
- [B10] **J. Zhuang.** *3D-Reconstruction of specular surfaces using CNNs.* IIIT, 2020.
- [B11] **F. L. Kühner.** *Rekonstruktion spiegelnder Oberflächen mittels monokularer Deflektometrie und künstlichen neuronalen Netzen.* IIIT, 2020.
- [B12] **F. Grauer.** *Mirror Surface Reconstruction by Specular Flow in Light Fields.* IIIT, 2020.
- [B13] **T. Zhao.** *Verbesserung der generalisierten Kamerakalibrierung durch Modellierung des Referenzmonitors.* IIIT, 2020.
- [B14] **M. Seeger.** *Lichtfeldbasierte Multi-Stereo-Deflektometrie mit einer generischen Kamerakalibrierung.* IIIT, 2021.