TÉCNICO
LISBOA

UNIVERSIDADE DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

Vision with Plenoptic Cameras

Nuno Miguel Barroso Monteiro

Supervisor: **Doctor José António da Cruz Pinto Gaspar**
Co-Supervisor: **Doctor João Pedro de Almeida Barreto**

Thesis approved in public session to obtain the PhD Degree in

Electrical and Computer Engineering

Jury final classification: Pass with Distinction

2021

UNIVERSIDADE DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

Vision with Plenoptic Cameras

Nuno Miguel Barroso Monteiro

Supervisor: **Doctor José António da Cruz Pinto Gaspar**

Co-Supervisor: **Doctor João Pedro de Almeida Barreto**

Thesis approved in public session to obtain the PhD Degree in Electrical and Computer Engineering

Jury final classification: Pass with Distinction

Jury

Chairperson: **Doctor João Manuel Lage de Miranda Lemos, Instituto Superior Técnico, Universidade de Lisboa**

Members of the Committee:

Doctor Bastian Goldluecke, Faculty of Computer and Information Science, University of Konstanz, Germany

Doctor Mário Alexandre Teles de Figueiredo, Instituto Superior Técnico, Universidade de Lisboa

Doctor José Alberto Rosado dos Santos Victor, Instituto Superior Técnico, Universidade de Lisboa

Doctor Nuno Miguel Mendonça da Silva Gonçalves, Faculdade de Ciências e Tecnologia, Universidade de Coimbra

Doctor José António da Cruz Pinto Gaspar, Instituto Superior Técnico, Universidade de Lisboa

Funding Institutions

Fundação para a Ciência e para a Tecnologia

2021

Abstract

Vision is one of the most important sensing modalities in nature because of the valuable, thorough information it can provide about the environment. Vision sensing can come in different flavors ranging from human vision, where images are perspective views that follow the pinhole model, to insect vision where compound eyes with ommatidia design enable the acquisition of multiview images of nearby objects which are highly effective to live and navigate in fast changing 3D environments. Recent technological advances allow mimicking this natural, multiview vision using plenoptic cameras. This thesis approaches plenoptic vision for the case of cameras that combine a single high-definition imaging sensor, a microlens array and a main lens.

The plenoptic camera does not follow the pinhole model that is broadly used in computer vision to describe the projection in conventional cameras that mimic the human eye. The plenoptic camera can be understood as a human eye where the retina is replaced by a compound eye, and where geometric and depth perception aspects deviate from what is taught in classical 3D computer vision. In this thesis is taken the constructive approach of leveraging classical projection models to represent plenoptic cameras as camera arrays that are familiar and intuitive to the average practitioner. State of the art calibration tools for plenoptic cameras are incorporated based on the proposed representation. New functionalities are added such as estimating disparities with differential operators.

The contributions of this work comprise (i) models that describe both standard and multifocus designs of the plenoptic camera in a common framework, (ii) a seminal study that analyzes the depth reconstruction capabilities of the standard plenoptic camera, (iii) new calibration methods that build on the proposed representation of the plenoptic camera as a camera array to estimate the calibration parameters in a linear, intuitive manner, and (iv) improvements on existing single image reconstruction methods based on intrinsic depth cues and on the concept of affine Lightfield (LF).

Keywords: Plenoptic Cameras, Camera Arrays, Calibration, 3D Reconstruction, Affine LF

Resumo

A visão é um dos sensores mais importantes na Natureza devido à valiosa e detalhada informação que fornece acerca do ambiente circundante. Esta capacidade sensorial abrange diversos sistemas visuais desde a visão humana, que adquire imagens de perspectiva seguindo o modelo *pinhole*, até à visão de insecto onde os olhos compostos por omatídeos permitem a aquisição de imagens *multiview* para objetos próximos que tornam eficaz a navegação num mundo 3D em constante mudança. Os recentes avanços tecnológicos permitem simular esta visão *multiview* usando câmaras plenóticas. Esta tese foca-se na visão plenótica para o caso de câmaras que incluem um sensor de imagem, um *array* de microlentes e uma lente principal.

A câmara plenótica não segue o modelo *pinhole* que é largamente utilizado em visão por computador para descrever a projeção de câmaras convencionais que simulam o olho humano. A câmara plenótica pode ser interpretada como um olho humano no qual a retina é substituída por um olho composto, e onde a geometria e a perceção de profundidade se desviam do que é classicamente ensinado em visão por computador. Nesta tese é seguida uma abordagem construtiva que parte de modelos de projeção clássicos para representar câmaras plenóticas como *arrays* de câmaras que são familiares e intuitivos. Usando a representação de *array* de câmaras, são propostos novos métodos de calibração para câmaras plenóticas. Adicionalmente, são apresentadas novas funcionalidades tais como a estimação de disparidade usando operadores diferenciais.

As contribuições deste trabalho compreendem (i) modelos que permitem descrever as variantes *standard* e multi-foco das câmaras plenóticas sob uma *framework* comum, (ii) um estudo seminal que analisa as capacidades de recons-trução da câmara plenótica *standard*, (iii) novas metodologias de calibração tendo por base o *array* de câmaras proposto para representar a câmara plenótica e que permitem estimar os parâmetros de calibração de forma linear e intuitiva, e (iv) melhorias aos métodos de reconstrução baseando-se em características de profundidade intrínsecas e no conceito de *lightfield* afim.

Palavras-Chave: Câmaras Plenóticas, Sistemas de Câmaras, Calibração, Re-construção 3D, LF Afim

Contents

List of Figures

List of Tables

Acronyms

ADMM Alternating Direction Method of Multipliers

DDFFNet Deep Depth From Focus Network

DLT Direct Linear Transformation

EPI Epipolar Plane Image

FFT Fast Fourier Transform

FOV Field of View

FPC Focused Plenoptic Camera

LF Lightfield

LFIM Lightfield Intrinsic Matrix

MI Microlens Image

MPC Multifocus Plenoptic Camera

MSE Mean Squared Error

RMS Root Mean Square

SALSA Split Augmented Lagrangian Shrinkage Algorithm

SCam Surface Camera Image

SPC Standard Plenoptic Camera

STD Standard Deviation

SVD Singular Value Decomposition

VI Viewpoint Image

List of Publications

Journals

J.I (Submitted) N. Monteiro, L. Palmieri, T. Michels, L. Cruz, R. Koch, N. Gonçalves, J. Gaspar, ”Geometric Calibration of Multifocus Plenoptic Cameras”, IEEE Transactions on Cybernetics, May 2020.

J.II N. Monteiro, J. Barreto, J. Gaspar, ”Standard Plenoptic Cameras Mapping to Camera Arrays and Calibration based on DLT”, IEEE Transactions on Circuits and Systems for Video Technology (TCSVT), vol. 30, pp. 4090-4099, November 2019.

J.III N. Monteiro, S. Marto, J. Barreto and J. Gaspar, ”Depth Range Accuracy for Plenoptic Cameras”, Computer Vision and Image Understanding (CVIU), vol. 168C, pp. 104-117, January 2018.

International Conferences

I.I N. Monteiro, J. Gaspar, ”Generalized Camera Array Model for Standard Plenoptic Cameras”, Fourth Iberian Robotics Conference (ROBOT), Porto, Portugal, 20-22 November 2019.

I.II N. Monteiro, J. Gaspar, ”Standard Plenoptic Camera Calibration for a Range of Zoom and Focus Levels”, Ninth Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA). Madrid, Spain, 1-4 July 2019.

I.III L. Dihl, L. Cruz, N. Monteiro, N. Gonçalves, ”A Content-Aware Filtering for RGBD Faces”, International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP). Prague, Czech Republic, 25-27 February 2019.

I.IV S. Marto, N. Monteiro, J. Barreto, J. Gaspar, ”Structure from Plenoptic Imaging”, IEEE International Conference on Developmental and Learning and on Epigenetic Robotics (ICDL-EpiRob). Lisbon, Portugal, 18-21 September 2017.

I.V N. Monteiro, J. Barreto, J. Gaspar, ”Dense Lightfield Disparity Estimation Using Total Variation Regularization”, International Conference on Image Analysis and Recognition (ICIAR), Póvoa de Varzim, Portugal, 13-15 July 2016.

I.VI N. Monteiro, J. Barreto, J. Gaspar, ”Influence of Positive Instances on Multiple Instance Support Vector Machines”, Second Iberian Robotics Conference (ROBOT), Lisbon, Portugal, 19-21 November 2015.

National Conferences

N.I D. Portela, N. Monteiro, J. Gaspar, ”Camera Adaptation for Deep Depth from Lightfields”, Portuguese Conference on Pattern Recognition (RecPad). Coimbra, Portugal, 26 October 2018.

N.II S. Marto, N. Monteiro, J. Gaspar, ”Locally Affine Lightfields as Direct Measurements of Depth”, Portuguese Conference on Pattern Recognition (RecPad). Coimbra, Portugal, 26 October 2018.

N.III N. Monteiro, J. Barreto, J. Gaspar, ”Surface Cameras from Shearing for Disparity Estimation on a Lightfield”, Portuguese Conference on Pattern Recognition (RecPad). Coimbra, Portugal, 26 October 2018.

Master Thesis

M.I Tiago Miguel Pereira Torres, ”Neuromorphic Image Reconstruction”, May 2019.

M.II Diogo Filipe Baptista Portela, ”Deep Depth from Plenoptic Images”, November 2018.

M.III Simão Pedro da Graça Oliveira Marto, ”Structure Reconstruction using Plenoptic Cameras”, November 2017.

Chapter 1

Introduction

The valuable information about the surrounding environment provided by vision makes it one of the most important sensors in Nature. Namely, it is estimated that three-quarters of the information in the human brain comes through the eyes [118]. Natural selection also exhibits the importance of vision, eyes (simple or complex) are present in almost 90 percent of all animal species [118] and evolved independently several times [85, 118]. This explains the diversity and highly specialized visual systems observed in Nature.

Human vision is extremely effective for the human life but, in various aspects, is surpassed by the vision of other animals. The mantis shrimp perceives 12 colors, as opposed to the three colors of the human visual system, some outside the visible spectrum of the electromagnetic radiation. This is a consequence of the 16 different types of photoreceptors in the mantis shrimp retina [94]. There are other visual systems specialized for low light conditions or for seeing more detail. More specifically, the shark has ten times better vision than humans in dark conditions due to a mirror like layer at the base of the retina that reflects the incoming rays [118]. The peregrine falcon, on the other hand, sees up to five times more detail than humans as a result of having two foveas in each eye [118].

The highly specialized visual systems in Nature and the advances in digital camera technology, optical fabrication and computational processing motivated the appearance of new types of cameras, like hyperspectral [24], event-based [45] or plenoptic cameras [111, 120], that exploit some particular property of the biological systems to improve man made systems. These cameras do not follow the computer vision conventional camera imaging that replicates the human visual system experience, *i.e.* capture a trichromatic two dimensional image.

In this thesis, one objective is leveraging our understanding of necessary brain functions for insects like flies effectively perceive the 3D world. This is aligned with the Robotics, Brain and Cognition program goals [1]. Our approach is based on studying a recent type of camera, the plenoptic camera [3, 113].

1.1 Motivation

The compound eyes found in insects and crustaceans are composed of multiple simple and tiny units called ommatidia (Figure 1.1). Each unit consists of a cornea, a lens and a small number of photoreceptors [19, 118] that observe a small region of the scene and generates images of low spatial resolution and high temporal resolution when compared to the human visual system [138]. Optical setups replicate the compound eye by placing a microlens array in front of the sensor [116]. The plenoptic cameras (Figure 1.1) studied in this thesis differ from these setups by presenting an additional main lens in front of the microlens array. In Nature, no animal has been discovered to possess such visual system [85], instead these cameras can be interpreted as a human eye in which the retina is replaced by a compound eye [113].

In insects such as dipterans (flies), understanding distance vision and depth perception is still insufficiently resolved [98]. Although stereoscopy has been found to be used by the praying mantis [124], it is still unclear if this system is also present in flies [19]. Namely, the two outward pointing compound eyes have almost no overlap and stereopsis requires a substantial overlap between the Field of View (FOV) of the two eyes [23]. The more frontal ommatidia have overlapping FOVs but

[1] Portuguese Roadmap of Research Infrastructures FCT 01/SAICT/2016 SAICT Proj. 22084.

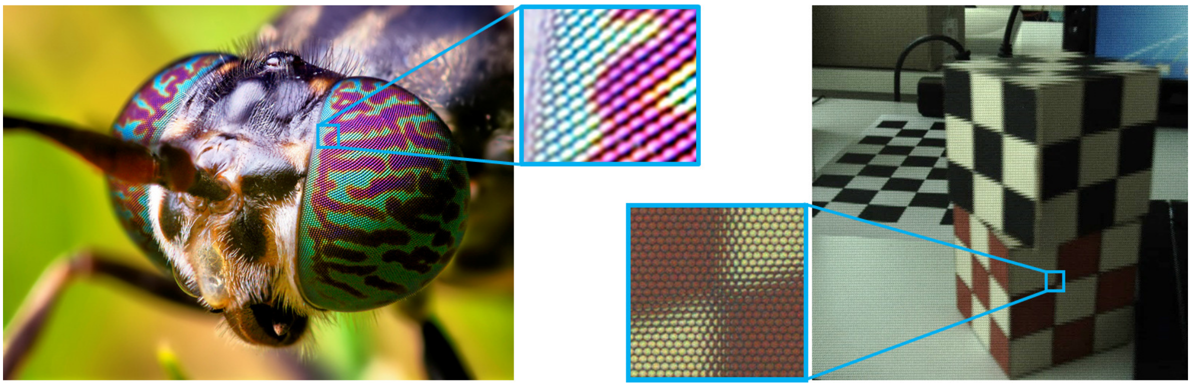


Figure 1.1: Black Soldier Fly Head with two compound eyes and detail of their ommatidia (left) (photo captured by Thomas Shahan and released under the Creative Commons License). The ommatidia of the compound eye resemble the MIs formed in the sensor of a plenoptic camera (right).

no neurons have been found to receive information from the two eyes, which would be how neural circuits process depth information as in the case of primates [6, 70, 121]. Additionally, the small distance between neighboring ommatidia of the individual eyes makes stereo not meaningful [19]. Therefore, other strategies must be in place for flies to perceive depth.

Plenoptic cameras allow to perceive the scene in different ways by reorganizing the pixels captured by each microlens [29, 112]. The compound eyes in flies, on the other hand, have a complex wiring such that all photoreceptors looking at the same point in visual space send information that converges upon the same synaptic unit. Consequently, even if the ommatidia are separate entities, the information is intermingled at the immediate next step in visual processing (neural superposition) [33, 81]. This allows to hypothesize that the information in different ommatidia can be combined to extract information of the scene as in plenoptic cameras. Indeed, in the work of Bitsakos and Fermüller [19], a first attempt is made to explain depth perception in flies using the Lightfield (LF). Thus, one hopes that the findings presented in this thesis can give new insights to process depth information and in this way allow neurophysiology, biology and engineering to grow together.

Despite having relevance in biology, in the recent years, LF has also gained importance in industry. The awareness of LF is being followed by the appearance of startups that develop optical setups for LF acquisition like Raytrix and K-Lens, LF displays for 3D-TV like Holografika or augmented reality like Creal3D. More mature industries and companies are starting to use LF to improve their products. Namely, in the mobile industry it is common to see nowadays multiple cameras on the rear of a mobile phone. These cameras are used together to improve the resolution, focus and lighting conditions of the final rendered image. Nonetheless, LF and plenoptic cameras require specific knowledge that limits their usability. More specifically, these cameras are described by camera models that are not built based on computer vision knowledge from decades ago. Hence, in this thesis, one additional, principal, objective is establishing the connection between the specific plenoptic camera knowledge and computer vision knowledge to make these cameras accessible to a broader range of users and applications.

1.2 Contributions

This thesis moves away from the mainstream by addressing imagery in which the projection model has multiple projection centers (non-central cameras). More specifically, are studied plenoptic cameras.

The composition of optical elements in a plenoptic camera allows to obtain interesting effects like a dynamic parallax. In a plenoptic camera, objects farther away from the camera can exhibit large parallax while closer objects can exhibit no parallax. This is contradictory to our standard notion of parallax that objects farther away exhibit almost no parallax. In commercial plenoptic cameras, this notion of parallax can be changed after the LF is acquired by resampling the rays collected. One specific objective of this thesis is the formalization of models and techniques to deal with this type of imagery.

Throughout the thesis and with the goal of making plenoptic cameras more accessible to a broader audience, focus is placed on answering: (i) which optical setups can be represented by the plenoptic camera model, (ii) what is an equivalent representation for plenoptic cameras, and (iii) how can one use this equivalent representation to improve plenoptic camera applications.

The contributions achieved throughout the course of this thesis are four-fold:

- **Unification of Geometric Projection Models.** There are different optical setups that can capture the LF [120, 113, 144]. Each different optical setup has its corresponding geometric projection model and even for the same optical setup there are different projection models [21, 41]. In this thesis, is shown that the camera arrays and the different plenoptic cameras can be represented using the same geometric projection model. The unifying model allows studying similarities and equivalences of plenoptic camera models proposed in the literature.

- **Camera Array-based Representation of Plenoptic Cameras.** The first acquisitions of LFs were obtained using a single moving camera [57, 86] or an array of cameras [144]. The type of images obtained by plenoptic cameras also allow to conveniently view them as camera arrays [140]. In this thesis, are studied the Standard Plenoptic Camera (SPC) and Multifocus Plenoptic Camera (MPC) and are proposed representations for these cameras based on the viewpoint or the microlens array, respectively. These representations are used to define new calibration methodologies and study the properties of the camera arrays.

- **In Depth Study of SPCs.** SPCs, as other plenoptic cameras, allow to obtain depth from a single image. Additionally, these cameras provide some interesting metadata parameters regarding their optical setup with the acquired images. Nonetheless, the depth capabilities of these cameras have not been evaluated and the metadata parameters provided are only used to initialize the decoding process

[32, 41]. In this thesis, is presented a first study regarding the SPC depth capabilities and a detailed analysis of the metadata parameters.

- **Depth Estimation Boosting and Refinement.** There are several strategies to recover the 3D information of the scene from LFs, usually based on correspondences and defocus cues [129, 135, 139]. However, these strategies only recover disparity information for the central viewpoint and can only retrieve reliable estimates on particular regions of the LF. Furthermore, the conversion between disparity and depth that is normally used for LF [22] cannot be applied to plenoptic cameras. In this thesis, are improved several disparity estimation techniques by considering the geometry of the camera arrays defined and by introducing the concept of affine LF.

1.3 Outline of the Thesis

In terms of structure, Chapter 2 overviews the major concepts behind plenoptic cameras. In Chapter 3, is described the mapping between rays in the image space and rays in the metric space found in the literature for plenoptic cameras, and is defined a ray-based projection model using this mapping.

The ray mapping has no connection with the pinhole projection matrix and the definition of their entries is rather complex. Hence, in Chapters 4 and 5, is defined the connection of the ray mapping with an array of viewpoint and microlens cameras described based on the pinhole projection matrix, respectively. The representation of the viewpoint cameras is used to propose calibration methodologies for SPCs while the representation of the microlens cameras is used to propose a calibration methodology for MPCs. In Chapter 4, are also evaluated the depth capabilities of SPCs.

The camera arrays described in Chapters 4 and 5 represent images whose ray collections can be readily obtained from plenoptic cameras.

Nonetheless, there are other ray collections that can be obtained from the rays acquired. In Chapter 6, one generalizes the camera array geometry to consider the different ray collections. Additionally, is extended the ray mapping to consider an array of plenoptic cameras. The camera array equivalent representation of an array of plenoptic cameras is used to propose a calibration methodology for this setup.

In Chapter 7, are explored depth reconstruction methodologies for plenoptic cameras and is introduced the concept of affine LF. This concept allows to define improvements and limitations of current methodologies found in the literature. In this chapter, is also proposed a dense reconstruction methodology to efficiently recover a depth estimate for each ray in the LF.

The major contributions and conclusions are summarized in Chapter 8. In this chapter, are also discussed some future lines of research for LF analysis and processing.

Chapter 2

Plenoptic Concepts

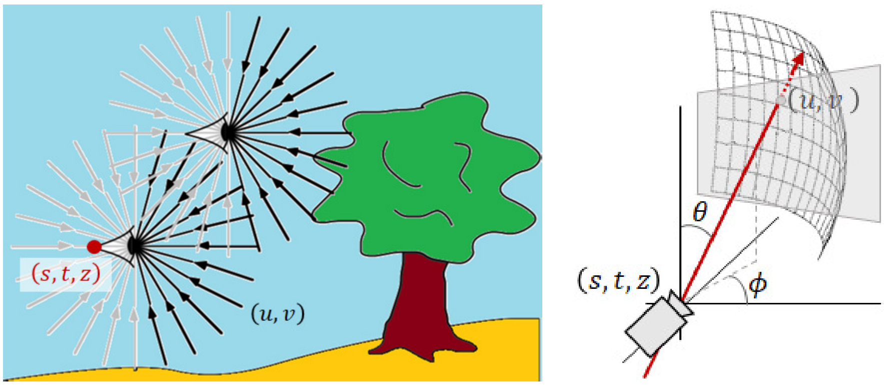
In this chapter, are overviewed the major concepts behind plenoptic cameras. More specifically, the chapter starts by describing the function that explains the distribution of light in space followed by the optical setups capable of sampling part of this function. The additional information captured by these optical setups opens new possibilities like refocusing or single image depth estimation as detailed in the next chapters.

2.1 Plenoptic Function

Cameras are capable of capturing the radiance of the light rays propagating from the scene. More specifically, these optical setups transform or acquire part of the information in the plenoptic funtion. The plenoptic function

$$\mathrm{L}(s, t, z, \theta, \phi, \lambda, t) \tag{2.1}$$

introduced by Adelson and Bergen [2] describes the total geometric distribution of light rays in space (Figure 2.1.a). This function gives the radiance of a light ray with wavelength λ and direction defined by the spherical coordinates (θ, ϕ) that passes through a point with spatial coordinates (s, t, z), viewer or camera projection center position, at an instant t. The direction of the light ray can be represented alternatively with the cartesian coordinates (u, v) defined by an imaginary plane at a unit distance from the viewer (Figure 2.1.b).



(a) Plenoptic Function (b) Parameterization

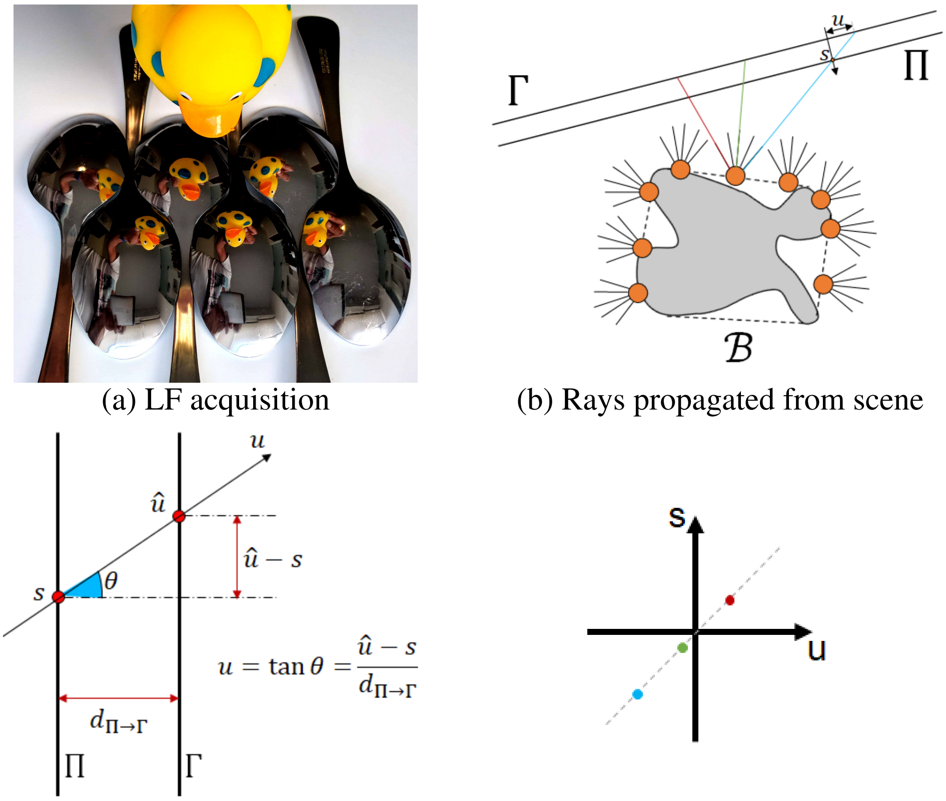
Figure 2.1: Plenoptic function representing the flow of light through space towards two observer locations **(a)** (adapted from [2]). The plenoptic function can be parameterized using spherical or cartesian coordinates **(b)**.

The plenoptic function is normally described using 7 parameters. While, the plenoptic function can be described using more parameters like polarization [31, 53], in this thesis, are considered only geometrical optics, *i.e.* rays are the fundamental elements for conveying light.

2.2 Lightfield

The acquisition of the full dimensionality of the plenoptic function is unfeasible, so one needs to consider some assumptions to reduce the dimensions being acquired. Nowadays, digital cameras allow to capture three independent color channels and acquire video. Hence, in practice, one is capable of acquiring six dimensions of the plenoptic function [120, 144]. In this thesis is considered a 4D function, the Lightfield (LF). The concept of the photic field [109], the lumigraph [57] or the LF [86] is similar to the concept of epipolar volumes [22] and can be traced back to the work of Gershun [55] describing the light radiometric properties in space.

The LF is a simplification of the 7D plenoptic function to a 4D function that describes the radiance of a light ray in its spatial and directional dimensions. Namely, considering static monochromatic light rays, the plenoptic function can be reduced to 5 dimensions, $\mathrm{L}(s, t, z, u, v)$. Considering that the rays are not attenuated or scattered and the viewer is outside the convex hull of the scene, *i.e.* in a region without a medium and occluders (free space), the plenoptic function can be further reduced to 4 dimensions $\mathrm{L}(s,t,u,v)$ since the radiance along the ray remains constant (Figure 2.2).


Γ
Π
u
s
$\mathcal{B}$
u
$\hat{u}$
θ
s
$\hat{u}-s$
$u = \tan\theta = \frac{\hat{u}-s}{d_{\Pi\rightarrow\Gamma}}$
$d_{\Pi\rightarrow\Gamma}$
Π
Γ
s
u

(a) LF acquisition

(b) Rays propagated from scene

(c) Ray parameterization

(d) Ray-space for (b)

Figure 2.2: LF acquisition, representation and parameterization using two parallel planes, and corresponding ray-space representation. **(a)** The spatial and directional information from the scene is acquired using an array of spoons while imaging a toy duck. **(b)** The LF assigns a radiance to each of the rays propagating from the scene and that are captured by the optical system. The boundary $\mathfrak{B}$ corresponds to the convex hull of the scene of interest. This boundary separates the free space that does not affect the light propagation from the scene of interest. **(c)** The intersection of a ray with the two planes Π and Γ define a geometry that allows to determine the direction of the ray. **(d)** Ray-space representation of the colored rays in (b). Notice that a line in ray-space define rays that intersect at a point in space.

The conditions to reduce the plenoptic function to the LF can also be considered inside the camera body [112]. The space inside the camera is considered a space free of a medium and occluders, so the rays inside the camera can be described solely using their intersection with the aperture inside the lens and the sensor plane (microlens plane considering lenticular array based plenoptic cameras).

In the following, is considered that the spatial dimension of the LF is associated with the different directions that are captured by a viewer (number of pixels in a camera) while the directional dimension is associated with the different viewer positions (number of camera positions). The LF, similarly to the plenoptic function, can have several parameterizations associated. The LF parameterization that is going to be used throughout this thesis is presented in Section 2.2.1. Additionally, a useful 2D representation of the LF, the ray-space, is described in Section 2.2.2.

2.2.1 Parameterization

The common parameterization for the LF describes the rays by their intersection with two parallel planes [57, 86] (Figure 2.2.b). This parameterization is obtained propagating the rays from the convex hull of the scene (boundary $\mathfrak{B}$). Hence, a light ray intersects the first plane Π at coordinates $(s, t, 0)$ and then intersects a second plane Γ at $(\hat{u}, \hat{v}, d_{\Pi\rightarrow\Gamma})$ (global two-plane parameterization). Considering that the planes are separated by a unitary distance ($d_{\Pi\rightarrow\Gamma} = 1$) and that the coordinates in plane Γ are defined relatively to (s, t), the ray $(s, t, \hat{u}, \hat{v})$ is defined as $(s, t, u = \hat{u} - s, v = \hat{v} - t)$. This parameterization is denoted as local two-plane parameterization and is equivalent to parameterizing the ray by a point (s, t) defined on plane Π and a direction (u, v) [71, 107] (Figure 2.2.c). This will be the parameterization considered throughout this thesis.

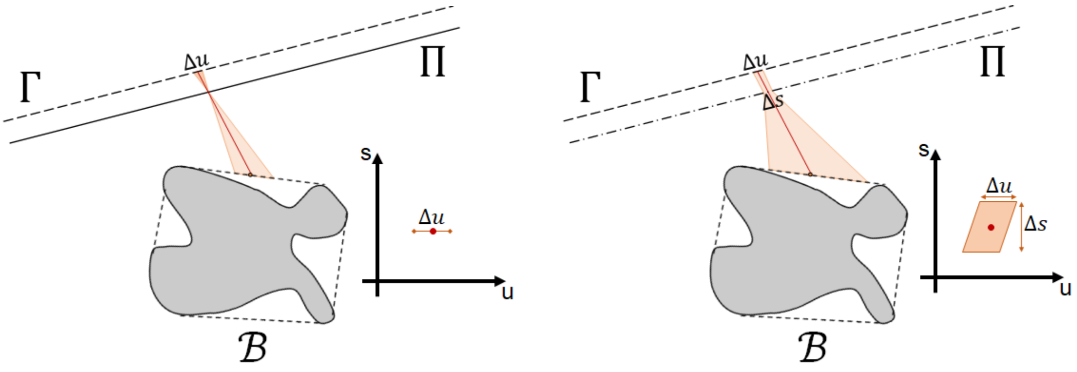
There are other representations that parameterize the rays on the surface of objects [30] (orange circles in Figure 2.2.b) but they are more complex and computationally expensive [112].

2.2.2 Ray-Space Representation

The cartesian ray-space is a two-dimensional space gathering information of a spatial (u) and a directional dimension (s) of the LF. In this space, a ray is represented by a point (s, u) and a set of rays is represented by a region. In particular, a line in ray-space represents the set of rays through a point in space [57] (Figure 2.2.d).

The illustration of the LF in two dimensions (s, u) makes analysis easier and straightforward. These analysis can be normally generalized to the complete LF in 4D (s, t, u, v).

The ray-space is useful as a tool to evaluate the LF sampling density [86] and the trade-off between spatial and directional resolution [54, 57]. Namely, a ray is only represented by a point (s, u) considering an infinitesimal pinhole and a non-discrete sensor. However, the rays are acquired by finite size pinhole apertures Δs and the corresponding radiance is recorded in sensors with finite size pixels Δu. Hence, a ray is represented by a sheared rectangle centered at (s, u) that describes the directional and spatial sampling of the LF (Figure 2.3).



(a) Ray-space with finite pixels

(b) Ray-space with finite pixels and pinhole apertures

Figure 2.3: LF sampling in ray-space representation considering finite size pixels **(a)** and finite size pinhole apertures **(b)**.

2.3 Lightfield Acquisition

In a conventional camera, the contribution of the light rays emanating from a given point in the scene is not distinguishable since they are averaged on a single pixel [133], *i.e.* the directional dimension of the LF is lost. The optical setups for acquiring the LF prevent the loss of information by describing the radiance of the light rays in the scene in its spatial and directional dimensions. This allows to discriminate the contribution of each of the point's light rays. The first idea of a device to capture the LF was illustrated by Mario Bettini [15] (Figure 2.4) and dates back to 1642. The illustration can be interpreted as a LF camera obscura.

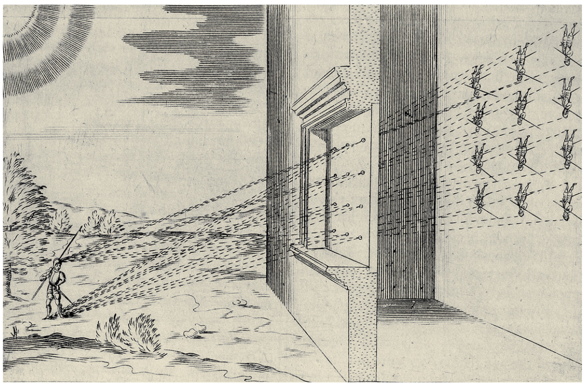
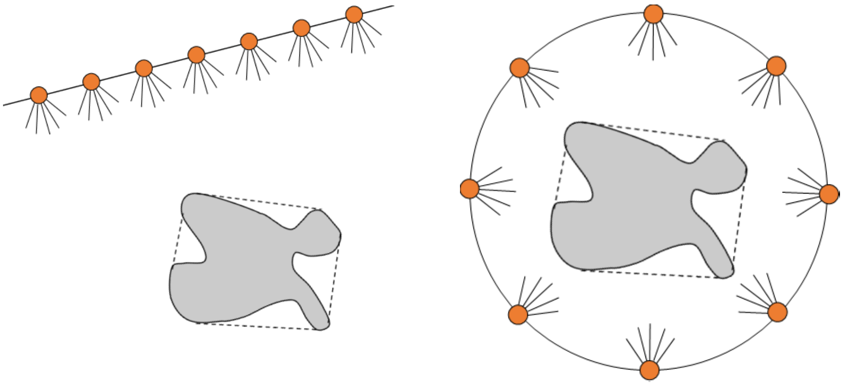


Figure 2.4: LF camera obscura by Mario Bettini [15]. Top of head and tip of one foot ray traced through an array of pinholes towards a projection plane, an indoor wall.

The devices for LF capture rely on multiple sensors or on a single sensor augmented by spatial or temporal multiplexing (Figure 2.5). The temporal multiplexing approach consists in a single moving camera taking pictures of a static scene from different viewpoints in different time instants [57, 86]. The camera can be moved along a planar surface or along a spherical surface always pointing at the center of the sphere [86]. One can also move the camera arbitrarily but this requires the camera pose to be estimated at each frame [57]. Instead of moving the camera, one may keep the camera static and move the objects in the scene accordingly [86], or one may keep the camera and scene static and rotate a planar mirror [72]. One of the difficulties with this approach is maintaining the same illumination conditions throughout the acquisition of the LF [86].

On the other hand, the spatial multiplexing approach consists in a single camera taking pictures of a scene from different viewpoints at the same time instant. This approach, contrarily to temporal multiplexing, allows to capture dynamic scenes. The equivalent setup relying on multiple sensors is the camera array [144, 145]. Nonetheless, camera arrays are expensive and require a complex positioning, alignment and synchro-



(a) Planar acquisition (b) Circular acquisition

Figure 2.5: Example of LF acquisitions. The planar **(a)** and the circular **(b)** acquisitions of the LF can be performed using a single moving camera (temporal multiplexing) or using multiple cameras.

nization of the cameras to reduce the post-processing steps. Thus, using a single sensor augmented with spatial multiplexing is a good alternative for LF acquisition.

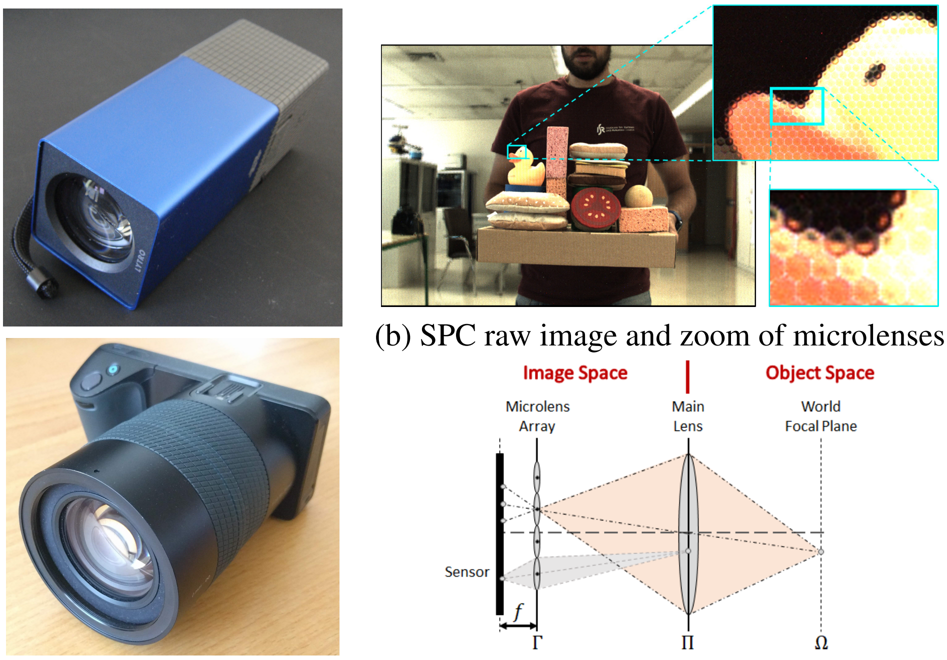
The idea of capturing the LF using a single sensor can be traced back to Lippmann in 1908 [50]. Lippman [50] proposed a sensitized plate with small spherical pieces of glass or other transparent material, resembling a rudimentary microlens array. Ives [75] proposed a similar setup using paralax barriers that act as multiple pinholes close to the sensor. Each of these cameras (lenses or pinholes) produce a small perspective view of the scene observed from that position of the array.

Adelson and Wang [3] coined the term plenoptic camera for their camera prototype, which was further enhanced to a portable hand-held plenoptic camera consisting of a sensor, microlens array, and main lens by Ng *et al.* [113]. This setup differs from the camera array by the narrow baseline and smaller spatial resolution [104]. These works extended the integral photography of Lippman [50] and the parallax panoramagram of Ives [74] to spatially multiplex the 4D LF onto a 2D image sensor. The limited size of the image sensor, limits the spatial and directional sampling defining a trade-off between the two types of information [54, 112]. For a more comprehensive explanation of the optical setups for LF acquisition, see Wetzstein *et al.* [143].

2.4 Plenoptic Cameras

In Section 2.3 were presented some of the different setups that can be used to acquire the LF. Although one can call the previous setups as plenoptic cameras, since they capture part of the plenoptic function, in this thesis, the setups consisting of a lenticular array [3, 91, 113, 120] are the ones termed as plenoptic cameras.

A lenticular array based plenoptic camera consists of a main lens, one single high-definition imaging sensor, and a microlens array. There are three types of lenticular array based plenoptic cameras, the Standard Plenoptic Camera (SPC), the Focused Plenoptic Camera (FPC) and the Multifocus Plenoptic Camera (MPC). These cameras differ on the microlenses and main lens' focal planes positioning and on the microlenses optical properties. The focal planes positioning influences the trade-off between spatial and directional information and the type of images produced by the microlenses.



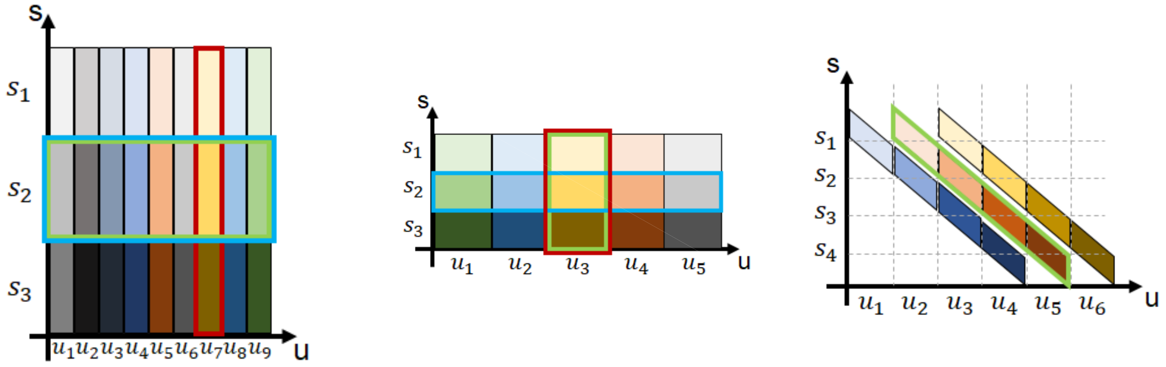
(a) SPCs

(b) SPC raw image and zoom of microlenses

(c) SPC geometry

Figure 2.6: SPC raw image and geometry. **(a)** Comercially available SPCs. **(b)** Image captured on the sensor of an SPC with detail of the MIs formed in the sensor. **(c)** Geometry of an SPC whose main lens focal plane corresponds to plane Ω.

The SPC [113] generates unfocused MIs by placing the main lens focal plane on the microlens array plane [52] (Figure 2.6). This allows to capture the radiance of the different directions, originating at a given point in the world focal plane of the main lens, in a microlens (orange). Similarly, the pixels beneath a microlens define the radiance of the point light rays observed from multiple viewpoints in the main lens aperture (gray) [54], considering the main lens to be at the optical infinity of the microlenses [1]. Hence, the optical configuration of an SPC allows to maximize the directional sampling of the LF arranging this information in the microlens pixels [112] (Figure 2.7.b).



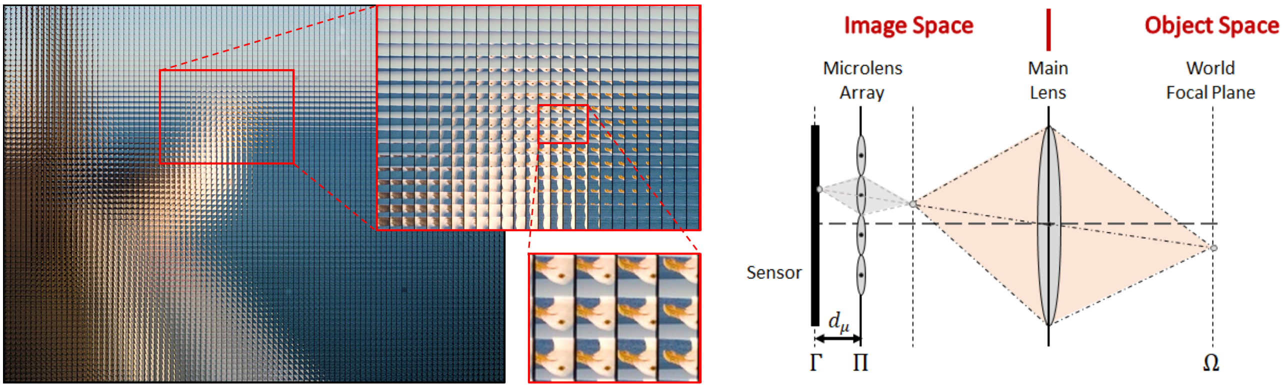
(a) Ray-space for camera array (b) Ray-space for SPC (c) Ray-space for FPC

Figure 2.7: Ray-space discretization and possible arrangements of the LF spatial and directional dimensions in the cameras' pixels (highlighted in green). In **(a)**, one represents the ray-space discretization for a camera array. The pixels sample the spatial dimension (u) while the cameras sample the directional dimension (s) of the LF. In **(b)**, the LF is sampled in the complete opposite way, *i.e.* the pixels and microlens cameras sample the directional and spatial dimension, respectively. This corresponds to the sampling performed on an SPC. In **(c)**, the sampling of an FPC is illustrated. These cameras offer a flexible trade-off between spatial and directional dimension by capturing in each pixel different spatial and directional information.

On the other hand, the FPC introduced by Lumsdaine and Georgiev [91] generates focused MIs by placing the focal plane of the microlenses on the main lens focal plane (Figure 2.8). Namely, a point in focus by the main lens will appear once in the microlens array while points out of focus will appear in more than one microlens [90]. Additionally, the out

[1] The microlenses are focused at infinity due to the positioning of the image sensor at their focal distance (f). The smaller size of the microlenses relatively to the distance between the microlens array and the main lens allows to assume that the main lens is at the optical infinity of the microlenses [112].

of focus points will appear blurred since the microlenses only focus on a single plane.



(a) FPC raw image and zoom of microlenses

(b) FPC geometry

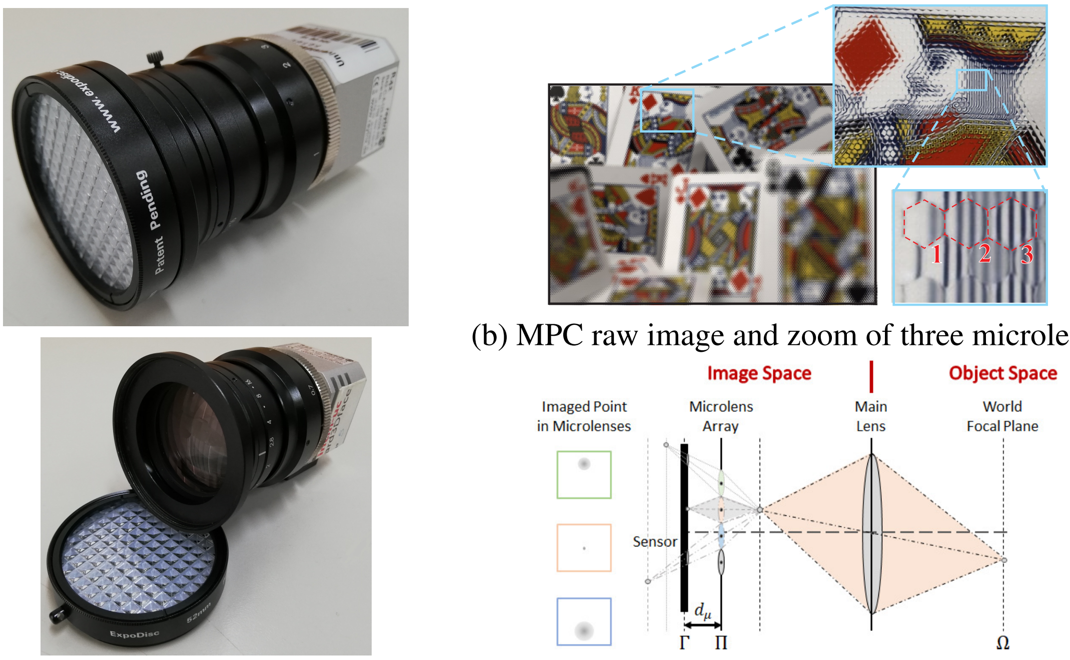
Figure 2.8: FPC raw image and geometry. **(a)** Image captured on the sensor of an FPC with detail of the MIs formed in the sensor. **(b)** Geometry of an FPC whose main lens focal plane corresponds to plane Ω.

The microlenses in an FPC act as micro-cameras (gray) sampling an image of the scene formed by the main lens inside the camera (orange). The FPC geometrical arrangement defines a flexible trade-off between the spatial and directional information of the LF [52, 112], capturing both dimensions intertwined on the microlens pixels (Figure 2.7.c). This allows to have a denser sampling of the LF spatial dimension relatively to the SPC [54, 91, 112].

The SPC and FPC comprise a microlens array with a single type of microlens. A different type of setup corresponds to the MPC [120] that has the same geometry of an FPC with a microlens array composed of different types of microlenses differing on their focal plane, *i.e.* focal length. Thus, the same scene point is imaged on each microlens type with different degrees of defocus (Figure 2.9).

2.5 Image Types

The light rays and the additional information captured in the LF allow to collect subsets of rays, one more straightforward than others, that define different images with different types of information and purposes. In this section, are described some of the image types that can be obtained



(a) MPCs

(b) MPC raw image and zoom of three microlenses

(c) MPC geometry, microlenses with three focal lengths

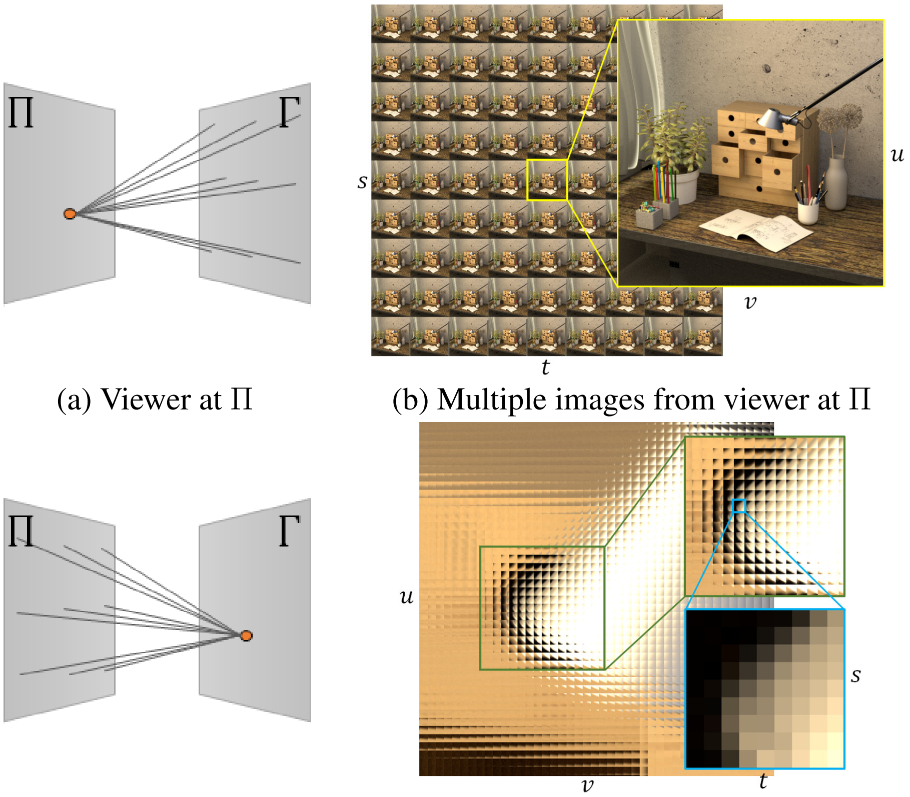
Figure 2.9: Mutifocus effect present in images acquired with MPCs **(a)**. **(b)** Image acquired by an MPC [5]. Small region is augmented to show microlens borders and focusing. MIs, 1 and 2 are blurred, 3 is focused. **(c)** MPC geometry illustrating the focused and blurred image formation.

from the LF.

2.5.1 Spatial and Directional Images

The independent sampling of the spatial and directional LF dimensions (Figure 2.7.a-b) allows to represent the acquired LF either as a collection of perspective images exhibiting a spatial view of the LF (highlighted in cyan in Figure 2.7) or as a collection of perspective images exhibiting the directional view of the LF (highlighted in red in Figure 2.7) [86]. Figure 2.10 shows the LF acquired by a planar camera array considering the two image collections described. Namely, one depicts the images obtained by each camera in the array positioned at plane Π (spatial view in Figure 2.10.a-b), and the reflectance map like images of the virtual cameras at plane Γ (directional view in Figure 2.10.c-d).

In lenticular array based plenoptic cameras, the LF is acquired on a single sensor. The image recorded on the sensor is denoted as raw image (Figure 2.11.a) and displays the images formed by each microlens in the



(a) Viewer at Π (b) Multiple images from viewer at Π

(c) Viewer at Γ (d) Multiple images from viewer at Γ

Figure 2.10: LF image types using the Table dataset [68]. The LF can interpreted as a sequence of images obtained from a viewer at Π **(a-b)** or from a viewer at Γ **(c-d)**.

physical microlens array placed in front of the sensor. The contents of the MIs depend on the geometrical configuration presented in Section 2.4. The different geometrical configurations change the position of the LF parameterization planes Π and Γ inside the camera. Namely, in an FPC the plane Π corresponds to the microlens array and the plane Γ corresponds to the image sensor. On the other hand, in an SPC the plane Π corresponds to the main lens aperture and the plane Γ to the microlens array.

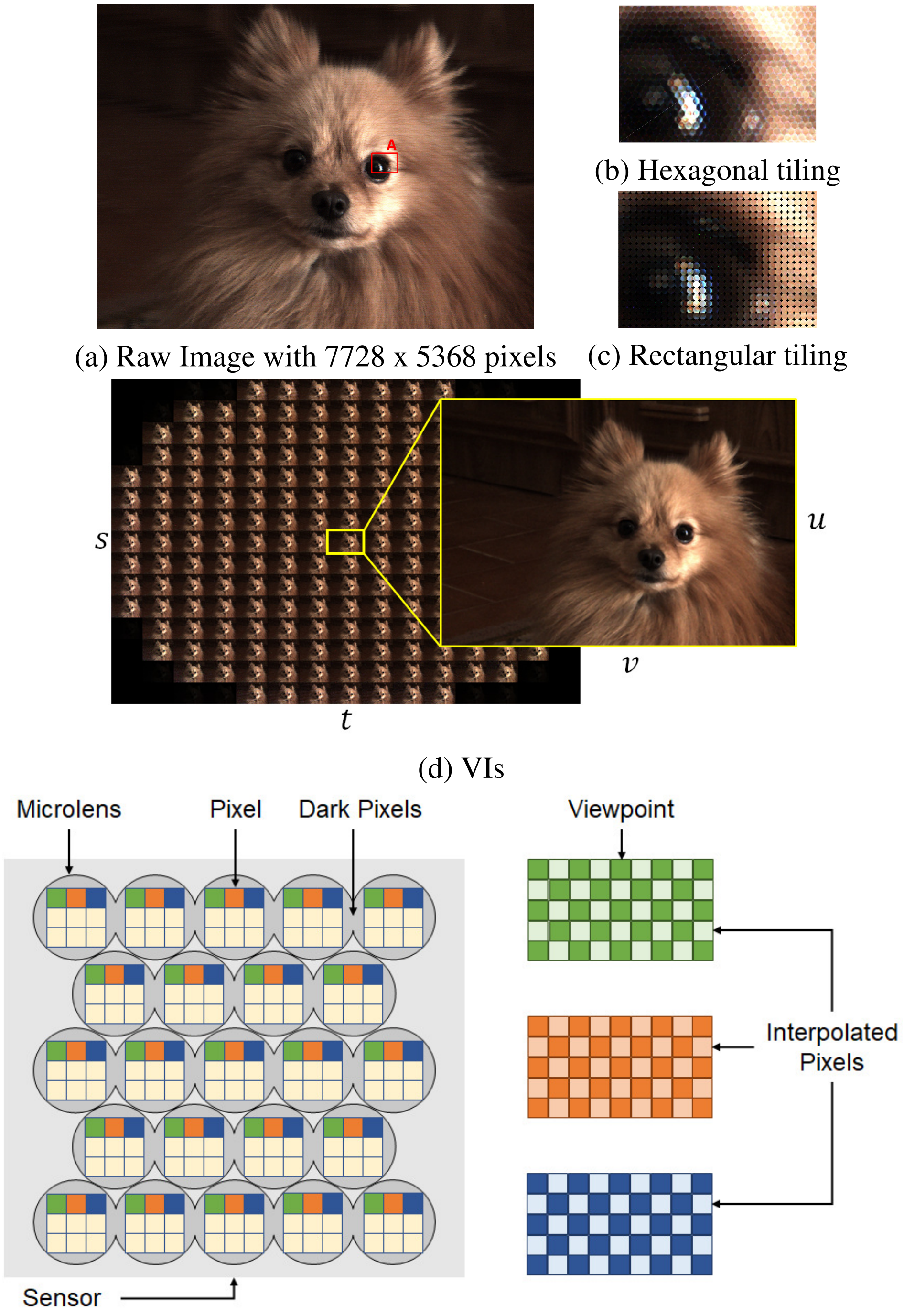
In an SPC, as in the camera array, one can define another arrangement of pixels that exhibits the spatial view of the LF. These images are denominated as viewpoint or sub-aperture images and are obtained by selecting and combining the same pixel position relatively to the microlens center for each microlens [112] (Figure 2.11). Thus, considering a microlens array having $P \times P$ microlenses with $N \times N$ pixels beneath each microlens, one can define $N \times N$ Viewpoint Images (VIs) having $P \times P$ pixels. This rearrangement defines a virtual camera array with coplanar projection centers and with a very narrow baseline [104]. These type of images are normally not considered for FPCs.

2.5.2 Epipolar Plane Images

The collection of images, microlens or viewpoints, define coplanar projection centers that are displaced horizontally and vertically between each other [21, 104]. Considering a subset of images, from one of the collections, such that the projection centers are equally spaced and define a linear path, one can define an Epipolar Plane Image (EPI) (Figure 2.12.b) like Bolles *et al.* [22] defined for a dense sequence of images acquired with a single moving camera.

The spatiotemporal nature of the EPIs described by Bolles *et al.* [22] is equivalently represented by the subset of images of the LF considering the spatial and the directional (temporal) dimensions placed on the horizontal and vertical axis, respectively. The EPI consists in collecting and stacking the epipolar lines from the subset of images on a single image. The process of creating EPIs is straightforward considering horizontal or vertical linear paths since the epipolar lines correspond to scan lines on the images. This is even more readily from the LF representation since the EPIs correspond to 2D slices, *i.e.* they can be obtained either fixing the coordinates (s, u) or (t, v).

In the EPIs, a point in the scene is projected onto a line whose slope describes the parallax and corresponds to the disparity of the point. This information can be used for reconstructing the scene from a LF (Figure 2.12.c-f). The approximately continuous baseline in plenoptic cameras allows computing the disparity using gradient operators in the EPIs [22, 39, 102] instead of using feature correspondences which improve the reconstruction process.



(a) Raw Image with 7728 x 5368 pixels

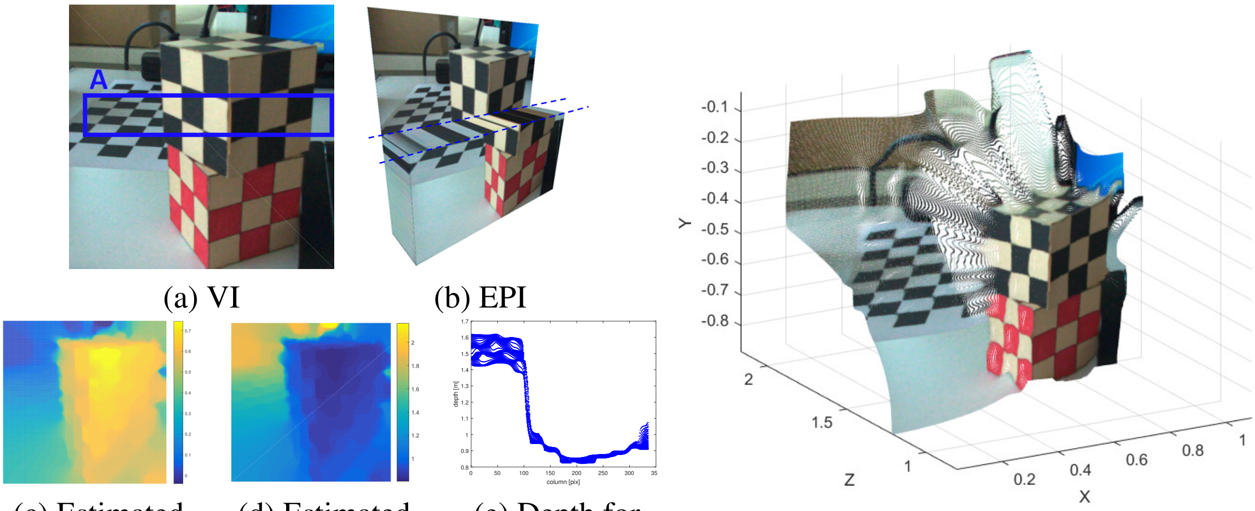
(b) Hexagonal tiling

(c) Rectangular tiling

(d) VIs

(e) Raw image and VIs schematics

Figure 2.11: Raw image **(a)** conversion to VIs **(d)**. The same pixel position in the different MIs is used to define a VI **(e)**. The hexagonal tiling **(b)** of the microlenses do not allow to fill all the pixels in a VI, therefore, during the decoding process [41], the missing pixels are interpolated (lighter colored pixels) and a rectangular tiling is obtained **(c)**.



(a) VI

(b) EPI

(c) Estimated disparity

(d) Estimated depth

(e) Depth for pixels in A

(f) Reconstructed point cloud

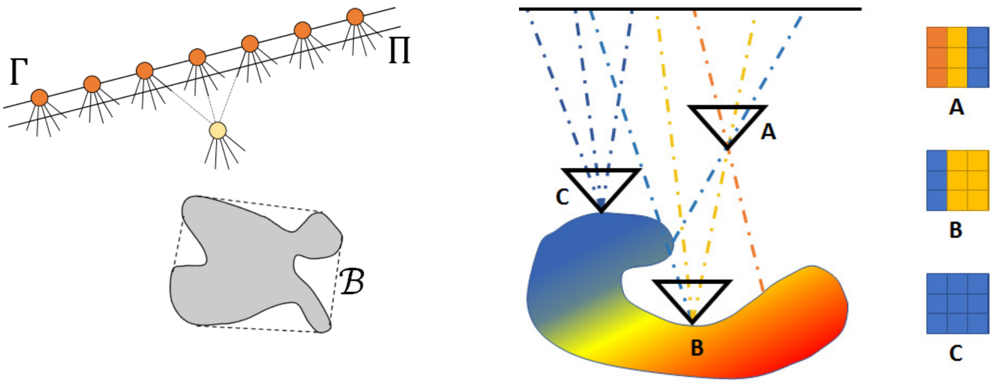
Figure 2.12: Depth reconstruction from the LF acquired by an SPC. The raw image allows to obtain 11×11 VIs with 378×379 pixels **(a)**. Considering an horizontal linear path, one can obtain the EPI **(b)**. Using gradient operators and regularization one can estimate disparities **(c)** that can be transformed to depth **(d)** and point cloud **(f)** in metric units using the camera intrinsic parameters. The depth values for the pixels in area A are shown in **(e)** sorted by column pixel number.

2.5.3 Surface Camera Images

The additional information provided by the LF allows to define other not so straightforward reorganization of pixels. This rearrangement of light rays allows to synthesize images considering arbitrary scene positions for the viewer [57, 86] (Figure 2.13). These images, denominated as Surface Camera Images (SCams), collect rays that can be used to identify correspondences, detect occlusions or surface characteristics [29, 147]. Namely, these rays emanate from different points if the intersection point (viewer) is located in free space (camera A of Figure 2.13.b) or is located on a surface point which is partially occluded (camera B of Figure 2.13.b). On the other hand, the rays emanate from a common point if the intersection point is defined on a surface point (camera C of Figure 2.13.b).

2.5.4 Refocused Images

The LF also allows to obtain images focused at different depths considering synthetic aperture photography [73, 86]. This approach simu-



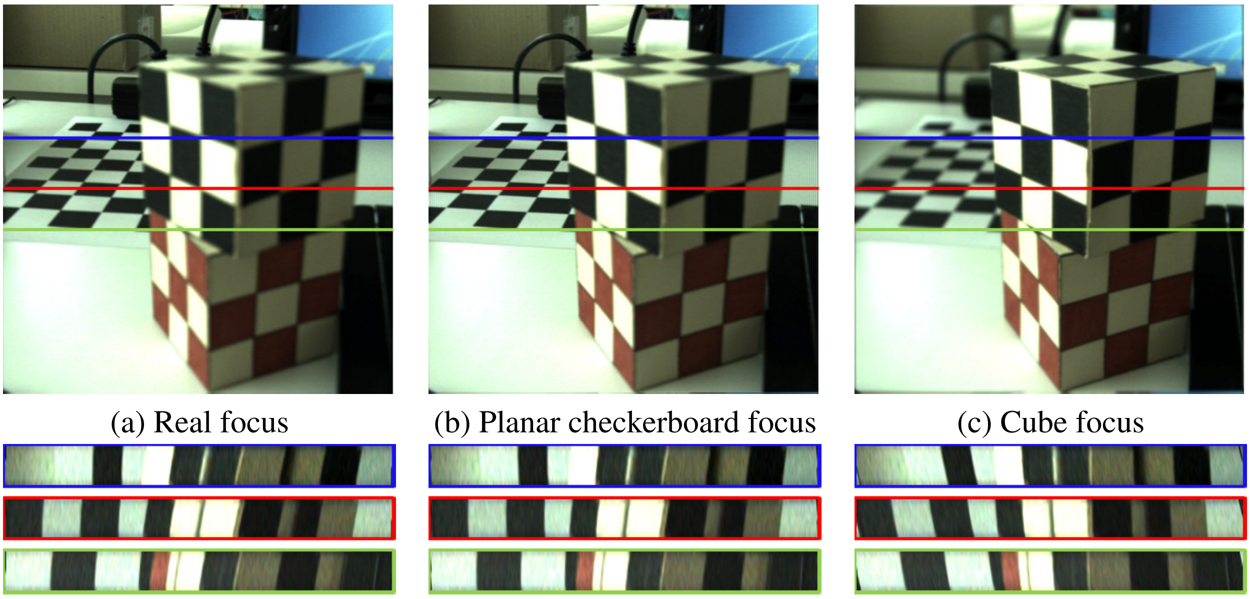
(a) SCam from LF (b) SCam types (adapted from Yu *et al.* [147])

Figure 2.13: SCams from LF. The additional information in the LF allows to rearrange the captured rays and define new views from arbitrary scene positions (yellow circle) **(a)**. According to the position of the rays' intersection points relatively to the surfaces in the scene, one can define several SCam types **(b)**.

lates the defocus blur by back-projecting the rays onto a real or virtual focal plane on the scene, and computing their average. The resulting image depicts sharp features for objects' surface points that intersect this plane (points in focus) while the surface points that do not intersect this plane are blurred (points out of focus) (Figure 2.14).

Considering the real focus plane, one can recover the image of the scene as if it was acquired by a conventional camera (Figure 2.14.a). More specifically, the conventional camera image can be obtained integrating the directional dimensions (s, t) of the LF in a process denominated as refocusing. The virtual focal plane is obtained applying the shearing operation in the LF before doing the refocusing of the LF [112] (Figure 2.14.b-c). The shearing operation allows to resample the EPIs in the original LF assuming a constant disparity (slope) at each pixel. The pixels on this disparity line of the original LF are considered to have disparity zero in the sheared LF (Figure 2.14.e-f).

The sampling of the LF determines the spatial resolution of the conventional camera image. The sampling performed by the SPC (Figure 2.7.b) determines that each microlens contributes with a single pixel for the conventional camera image [54]. Therefore, the size of the rendered image depends on the number of microlenses and will have a final resolution much lower than that of the image sensor [54, 91, 113]. Namely,



(a) Real focus (b) Planar checkerboard focus (c) Cube focus

(d) EPIs from original LF (e) EPIs from sheared LF with disparity 0.2 (f) EPIs from sheared LF with disparity 0.8

Figure 2.14: The LF allows to change the world focal plane by shearing the LF followed by refocusing. These images exhibit the original world focal plane **(a)** and two virtual focal planes. The virtual focal planes are placed at the planar checkerboard **(b)** and in the region of the cubes **(c)**. The EPIs **(d-f)** at rows 130 (blue), 180 (red) and 215 (green) for the different focal planes are depicted to highlight the shearing of the LF. The regions with vertical lines in the EPIs will appear focused while the other regions will appear blurred.

considering a Lytro Illum camera with 15×15 pixels in each microlens, one has 225 pixels in the sensor devoted to the directional sampling of the LF that can be used to obtain only one pixel in the rendered image.

On the other hand, the FPC (Figure 2.7.c) has a finer spatial sampling of the LF. More specifically, each microlens can contribute to the conventional camera image with a patch of pixels which allows to obtain a rendered image with higher spatial resolution [52]. The spatial resolution of the rendered image depends on the pixels in the MIs and their overlap [91]. However, the refocusing process is more complex involving integration across MIs since the directional samples of a spatial point are in different microlenses [52].

2.6 Chapter Summary

In this chapter was described the 7D plenoptic function and its simplification to the 4D LF. The LF defines a set of rays that are parameterized by a position (s,t) and a direction (u,v). These rays can be acquired using multiple cameras or a single camera augmented by temporal or spatial multiplexing. In this last approach, one incorporates the lenticular array based plenoptic cameras that can be divided in three different types: the SPC, the FPC and the MPC.

The spatial and directional dimensions of the LF allows to rearrange the rays captured into new images that enable applications like refocusing or depth estimation.

In the next chapters, are described in more detail the mapping of the captured light rays and the geometry of the different types of images that can be obtained from the LF captured by a plenoptic camera.

Chapter 3

Plenoptic Ray Geometry

This chapter details the geometry and the mapping of the rays captured by a Standard Plenoptic Camera (SPC) and a Focused Plenoptic Camera (FPC) and the rays in the object (metric) space. Using this mapping and the representation of a ray in the object space, is defined a ray-based projection model for a plenoptic camera.

3.1 Lightfield Intrinsic Matrix

In Chapter 2, were described the rays in the Lightfield (LF) using a point (s, t) defined on a parameterization plane Π and a direction (u, v) in metric units (Figure 3.1), *i.e.* the LF in the object space. However, the raw images obtained by plenoptic cameras exhibit pixels distributed by the corresponding microlens cameras (Figure 2.11.a-b). Hence, one can define an equivalent LF describing the rays using pixels (i, j) and microlens (k, l) indices, *i.e.* the LF in the image space. In this section, is presented the mapping between the LF in the object (metric) space and the LF in the image space defined by Dansereau *et al.* [41].

The mapping proposed by Dansereau *et al.* [41] is obtained by propagating the rays from the sensor to the object space using ray transfer matrices considering the main lens as a thin lens and the microlenses as pinholes (Section 3.5). In formal terms, the 5×5 matrix $\mathbf{H}$ is a mapping of back-projection rays $\tilde{\mathbf{\Phi}} = [i, j, k, l, 1]^T$ in the image space to rays $\tilde{\mathbf{\Psi}} = [s, t, u, v, 1]^T$ in the object space (Figure 3.1):

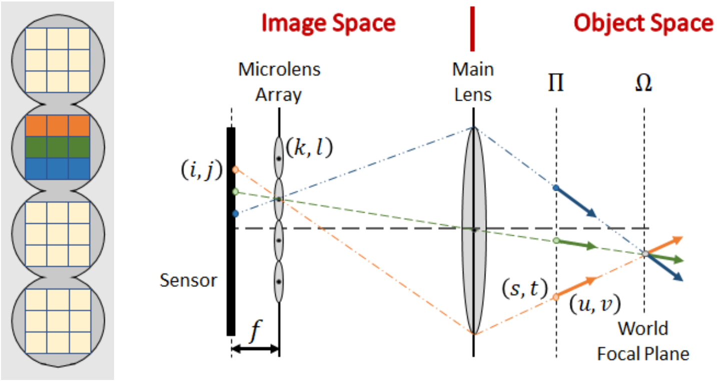


Figure 3.1: Mapping from the LF in the image space to the LF in the object space. The LF in image space is parameterized by microlens and pixel indices while the LF in object space is parameterized by a point and a direction. The mapping proposed by Dansereau *et al.* [41] defines an arbitrary position for the parameterization plane Π. Hence, different pixels in a microlens define different points and directions. If the parameterization plane is appropriately chosen, for example different pixels in a microlens can have a common point and define different directions (planes Π and Ω coincide).

$$\tilde{\mathbf{\Psi}} = \mathbf{H}\ \tilde{\mathbf{\Phi}} \tag{3.1}$$

where $\tilde{(\cdot)}$ denotes the vector $(\cdot)$ in homogeneous coordinates. In the following, this mapping will be denominated as the Lightfield Intrinsic Matrix (LFIM) [1] and consists of 12 non-zero parameters (10 free intrinsic parameters since h_s and h_t are fixed):

$$\mathbf{H} = \begin{bmatrix} h_{si} & 0 & h_{sk} & 0 & h_s \\ 0 & h_{tj} & 0 & h_{tl} & h_t \\ h_{ui} & 0 & h_{uk} & 0 & h_u \\ 0 & h_{vj} & 0 & h_{vl} & h_v \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} . \tag{3.2}$$

3.2 Related Work

The several works on plenoptic cameras consider the microlenses as pinholes and the main lens as a thin lens regardless of the type of plenoptic camera. One can divide the camera models in the literature in 2D-

[1] Note that LFIM is a simplified term, as $\mathbf{H}$ effectively contains intrinsic parameters information, however, it also contains baseline information, as detailed in [104] and [108]. Conventional extrinsic parameters, as found in pinhole camera models, defining a world coordinate system, are in fact not contained in $\mathbf{H}$.

based and 4D-based mappings. The 2D-based mappings describe the projection of a point in the object space on a particular camera, *i.e*. give the relationship between a point and a pixel. The 4D-based mappings describe the projection of the rays originating at a point in the object space onto a collection of 4D rays in the image space.

2D-based Mappings. Johannsen *et al*. [77] and Zeller *et al*. [148] proposed a Multifocus Plenoptic Camera (MPC) camera model using a single microlens type. In these works, the Microlens Image (MI) center is assumed to lie on the optical axis of the corresponding microlens which causes inaccuracy on the reconstructed points [60]. Additionally, Strobl *et al*. [132] noticed that the camera model of an MPC should consider the different microlens types. Heinze *et al*. [66] used a similar model to [77] accounting for the tilt-shift of the main lens and the different microlens types but not considering an end-to-end image formation.

Bok *et al*. [21] proposed an SPC camera model that describes a microlens camera using a projection matrix with 6 parameters and the knowledge of the corresponding microlens center in the raw image. Nousias *et al*. [115] showed that [21] can be extended to an FPC and considered this to describe an MPC as a collection of three independent FPCs. Nousias *et al*. [115] acknowledged the existence of common extrinsics among the microlens types but has not proposed a camera model combining these parameters which lead to a model with a high number of parameters to estimate.

4D-based Mappings. The mapping of rays defined in pixels (i, j) and microlenses (k, l) indices to rays defined by a position (s, t) and a direction (u, v) in metric units was first proposed by Dansereau *et al*. [41] for SPCs. More specifically for a virtual plenoptic camera whose microlenses define a rectangular tiling instead of the actual hexagonal tiling in the raw image (Figure 3.6.b). This virtual plenoptic camera is obtained after a decoding process to transform the 2D raw image into a 4D LF that is out of the scope of this thesis. For a better understanding of this process, the reader should refer to [41, 44]. This mapping considers a 5×5 matrix, the LFIM, with 10 free intrinsic parameters that is obtained by propagating the rays from the sensor to the object space using ray transfer matrices. In this thesis, one represents the LFIM with 8 free intrinsic parameters by shifting the rays parameterization plane along the optical axis of the camera [17] to the plane containing the viewpoint projection centers and removing the parameters redundant with the extrinsic parameters (Chapter 4) [104]. A similar representation can be obtained by considering the rays parameterization plane on the plane containing the microlens projection centers (Chapter 5) [108].

The LFIM was then generalized for FPCs [150]. Namely, Zhang *et al.* [150] proposed a generalized model that considers a LFIM with 6 free intrinsic parameters that is capable of representing the virtual SPC and the FPC. In this thesis, one shows that the model proposed by Zhang *et al.* [150] in fact corresponds to a 4D mapping with 8 free intrinsic parameters with 2 intrinsic parameters included in the radial distortion model (Appendix A). In the FPC, the LFIM describe the hexagonal tiling of the microlenses keeping the same structure of the SPC considering different sampling basis for the microlens coordinates [108] (Section 3.5).

The models presented in the literature for the 4D mapping only consider one microlens type. In this thesis, a 4D-based mapping is complemented with a blur model to describe the defocus behavior of each microlens type in an MPC (Chapter 5). This allows to extend the LFIM to an MPC and consider common intrinsic and extrinsic parameters among the microlenses types. Moreover, in this thesis, is extended the 4D-based mapping to a 6D-based mapping to represent a coplanar plenoptic camera array with the same world focal plane (Chapter 6).

Different plenoptic camera designs gave rise to various, specialized, geometric camera models [21, 41, 148]. Works [115, 150] and this thesis generalized these models to the different plenoptic cameras but almost no works established relationships between the different camera models and more specifically between the 2D-based and the 4D-based mappings. In this thesis, the different models are studied under a common framework (general model). This general model allows to represent a plenoptic camera despite the different calibration procedures for SPCs, FPCs and MPCs.

The definition of the projection matrices for the microlens and the viewpoint cameras of a plenoptic camera appeared in the work of Bok *et al.* [21]. The geometry of the camera arrays is described using the parameters of the optical setup and the knowledge of the corresponding microlenses centers in the raw image but no relationship with the original LFIM [41] is provided. Additionally, the geometry proposed for the viewpoint cameras assumes identical cameras which does not explain the zero disparity for points in the world focal plane of the main lens.

Marto *et al.* [95] (Appendix B) established a first connection between the mappings by representing a coplanar camera array composed of cameras with identical intrinsic parameters using a LFIM identical to the one from Zhang *et al.* [150]. Nonetheless, the camera arrays defined by an SPC or an FPC are not composed of identical cameras. More specifically, in this thesis, is shown that the 4D mapping of a plenoptic camera can be transformed to a 2D mapping to represent the virtual viewpoint camera array (Chapter 4) [104] and the physical microlens camera array (Chapter 5) [108]. These camera arrays consider coplanar cameras with a shifted principal point among the different cameras. Conversely, one shows that the model proposed by Bok *et al.* [21] can be represented by a 4D-based mapping constraining the microlenses centers coordinates on the raw image to be regularly spaced. This gives further confirmation that a 4D mapping can be extended to model an FPC and MPC.

Finally, is extended the characterization of the 2D-based mappings of the microlens and viewpoint camera arrays from the LFIM [21, 104, 108] and detailed the geometry of the several cameras that can be defined by collecting the rays captured by a plenoptic camera that intersect in an arbitrary point in the object space (Chapter 6).

3.3 Ray Parameterization

Let us consider a LF in the object space $L_\Pi(q,r,u,v)$ acquired by a plenoptic camera with the plane Ω in focus (Figure 3.2). The LF $L_\Pi(q,r,u,v)$ collects a set of rays where each ray $\tilde{\mathbf{\Psi}}_\Pi = [q,r,u,v,1]^T$ is parameterized using a point (q,r) defined on a parameterization plane Π and a direction (u,v) defined in metric units [107]. This parameterization allows to propagate the position in the ray originated at $[q,r,0]^T$ to an arbitrary plane at a distance λ using $[q,r,0]^T + \lambda\,[u,v,1]^T$, $\lambda \in \mathbb{R}$.

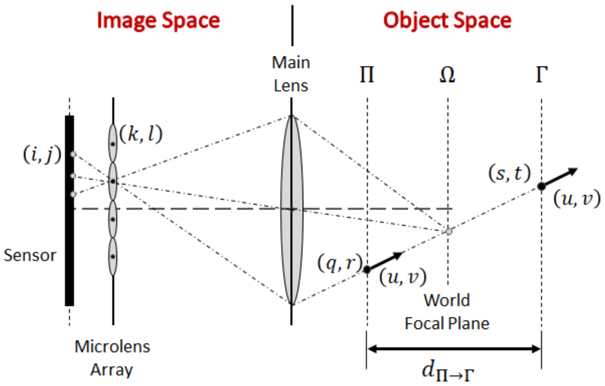


Figure 3.2: Ray parameterization in an SPC. The LF in the object space can be parameterized on an arbitrary plane regardless of the original plane Ω in focus.

3.3.1 Ray Re-Parameterization

The LF in the object space $L_\Pi(q,r,u,v)$ can be redefined on another plane Γ by shifting the parameterization plane Π along the optical axis of the plenoptic camera, *i.e.* along the normal to the plane Π (Figure 3.2). Assuming that Γ is at a distance $d_{\Pi\to\Gamma}$ from Π, the re-parameterization [17] is defined as

$$\tilde{\mathbf{\Psi}}_\Gamma = \mathbf{D}_r\,\tilde{\mathbf{\Psi}}_\Pi \tag{3.3}$$

where

$$\mathbf{D}_r = \begin{bmatrix} 1 & 0 & d_{\Pi\to\Gamma} & 0 & 0 \\ 0 & 1 & 0 & d_{\Pi\to\Gamma} & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} . \tag{3.4}$$

Note that $\mathbf{D}_r$ maps a ray $\tilde{\mathbf{\Psi}}_\Pi$ to a ray $\tilde{\mathbf{\Psi}}_\Gamma = [s, t, u, v, 1]^T$ representing a ray passing through a point (s, t) on plane Γ with a direction (u, v). Notice that $\mathbf{D}_r$ changes the camera coordinate system origin but does not change the directions (u, v).

3.3.2 Ray Parameterization Conversion

The LF in the object space considers rays parameterized using a point and a direction. However, one can represent the ray by its intersection with two planes (Section 2.2.1). Namely, defining the ray in the object space $\tilde{\mathbf{\Psi}}_{\Pi,\Gamma} = [q, r, s, t, 1]^T$ using a point (q, r) on plane Π and a point (s, t) on plane Γ, the parameterization change from the rays $\tilde{\mathbf{\Psi}}_\Pi$ is defined as

$$\tilde{\mathbf{\Psi}}_{\Pi,\Gamma} = \mathbf{D}_p \tilde{\mathbf{\Psi}}_\Pi \tag{3.5}$$

where

$$\mathbf{D}_p = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & d_{\Pi\to\Gamma} & 0 & 0 \\ 0 & 1 & 0 & d_{\Pi\to\Gamma} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} . \tag{3.6}$$

Contrarily to the re-parameterization (3.3), the matrix $\mathbf{D}_p$ does not change the camera coordinate system origin. Notice that the point (s, t) is defined considering a global coordinate system whose origin is defined on plane Π.

The ray parameterization of the LF in the object space $\tilde{\mathbf{\Psi}}_{\Pi,\Gamma}$ can be changed back to a parameterization using a point and a direction, assuming that the distance $d_{\Pi\rightarrow\Gamma}$ between the planes is known, considering

$$\tilde{\mathbf{\Psi}}_{\Pi} = \mathbf{D}_p^{-1}\tilde{\mathbf{\Psi}}_{\Pi,\Gamma} \tag{3.7}$$

where

$$\mathbf{D}_p^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ -1/d_{\Pi\rightarrow\Gamma} & 0 & 1/d_{\Pi\rightarrow\Gamma} & 0 & 0 \\ 0 & -1/d_{\Pi\rightarrow\Gamma} & 0 & 1/d_{\Pi\rightarrow\Gamma} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad . \tag{3.8}$$

Mapping the LF in the object space $L_{\Pi}\left(q,r,u,v\right)$ to the LF in the image space $L\left(i,j,k,l\right)$ using the LFIM $\mathbf{H}_{\Pi}$ by (3.1), one has

$$\tilde{\mathbf{\Psi}}_{(\cdot)} = \mathbf{D}_{(\cdot)}\ \mathbf{H}_{\Pi}\ \tilde{\mathbf{\Phi}} \quad . \tag{3.9}$$

The LFIM $\mathbf{H}_{(\cdot)} = \mathbf{D}_{(\cdot)}\ \mathbf{H}_{\Pi}$ maps the LF in the image space $L\left(i,j,k,l\right)$ to the re-parameterized LF in the object space $L_{\Gamma}\left(s,t,u,v\right)$ (3.4) or to the LF in the object space $L_{\Pi,\Gamma}\left(q,r,s,t\right)$ (3.6) where rays in the object space are parameterized using two points.

3.4 Ray-based Projection Model

One ray $\mathbf{\Psi} = \left[s,t,u,v\right]^T$ in the object space parameterized by a point (s,t) on plane Π and a direction (u,v) (Figure 3.1) can be represented as one parametric 3D line [58], namely

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} s \\ t \\ 0 \end{bmatrix} + \lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \ , \ \lambda \in \mathbb{R} \quad . \tag{3.10}$$

This equation allows to propagate the position in the ray originated at $[s, t, 0]$ to an arbitrary plane at distance λ from the origin of the camera coordinate system. Note that (3.10) generalizes a normalized pinhole camera: by setting $s = 0$, $t = 0$ and $(u, v) \in \mathbb{R}^2$ one obtains a pencil of lines. Therefore, by allowing $(s, t) \in \mathbb{R}^2$, one can represent an infinite number of normalized pinhole cameras.

Mapping the ray in the object space using (3.1), one defines the relationship between an arbitrary point $\mathbf{m} = [x, y, z]^T$ in the object space and the ray $\boldsymbol{\Phi}$ in the image space [107] as

$$\begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{H}^{st}_{ij} \begin{bmatrix} i \\ j \end{bmatrix} + \mathbf{H}^{st}_{kl} \begin{bmatrix} k \\ l \end{bmatrix} + \mathbf{h}_{st} + z \left(\mathbf{H}^{uv}_{ij} \begin{bmatrix} i \\ j \end{bmatrix} + \mathbf{H}^{uv}_{kl} \begin{bmatrix} k \\ l \end{bmatrix} + \mathbf{h}_{uv} \right) \tag{3.11}$$

where the LFIM $\mathbf{H}$ (3.2) is partitioned in four 2×2 diagonal sub-matrices

$$\mathbf{H}^{st}_{ij} = \begin{bmatrix} h_{si} & 0 \\ 0 & h_{tj} \end{bmatrix}, \; \mathbf{H}^{st}_{kl} = \begin{bmatrix} h_{sk} & 0 \\ 0 & h_{tl} \end{bmatrix}, \tag{3.12}$$

$$\mathbf{H}^{uv}_{ij} = \begin{bmatrix} h_{ui} & 0 \\ 0 & h_{vj} \end{bmatrix}, \; \mathbf{H}^{uv}_{kl} = \begin{bmatrix} h_{uk} & 0 \\ 0 & h_{vl} \end{bmatrix}, \tag{3.13}$$

and two 2×1 vectors $\mathbf{h}_{st} = [h_s, h_t]^T$ and $\mathbf{h}_{uv} = [h_u, h_v]^T$. Equation (3.11) shows that given one ray in image coordinates, the LFIM $\mathbf{H}$ allows defining a back-projection ray in the object space or, equivalently, one 3D point at a specific depth z.

Rewriting (3.11) relatively to (k, l), one can represent the ray-based projection model for a point $\mathbf{m}$ by

$$\begin{bmatrix} k \\ \\ l \end{bmatrix} = \begin{bmatrix} \mathrm{f}\left(i;\mathbf{m},\mathbf{H}\right) \\ \\ \mathrm{g}\left(j;\mathbf{m},\mathbf{H}\right) \end{bmatrix} = \begin{bmatrix} -i\,\dfrac{h_{si}+z\,h_{ui}}{h_{sk}+z\,h_{uk}} + \dfrac{x-h_s-z\,h_u}{h_{sk}+z\,h_{uk}} \\ \\ -j\,\dfrac{h_{tj}+z\,h_{vj}}{h_{tl}+z\,h_{vl}} + \dfrac{y-h_t-z\,h_v}{h_{tl}+z\,h_{vl}} \end{bmatrix} . \qquad (3.14)$$

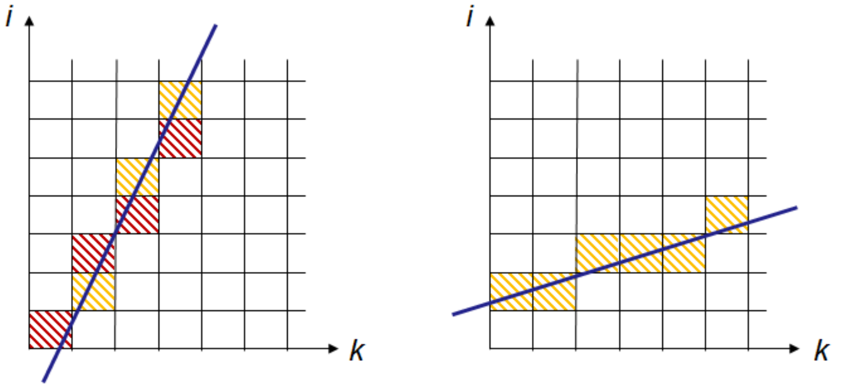
Note that $\mathrm{f}\left(i;\mathbf{m},\mathbf{H}\right)$ and $\mathrm{g}\left(j;\mathbf{m},\mathbf{H}\right)$ are mappings from $\mathbb{R} \to \mathbb{R}$, affine on the variables i and j. Since the point $\mathbf{m} \in \mathbb{R}^3$ and the LFIM $\mathbf{H} \in \mathbb{R}^5 \times \mathbb{R}^5$, the coordinates of the LF in the image space (i,j,k,l), in general cannot be all integers. Equation (3.14) shows that a point in the object space defines lines on the ray-spaces defined by each pair of coordinates (i,k) and (j,l) (Section 2.2.2).

Unlike common projection problems, as in the pinhole camera model, in a plenoptic camera a point $\mathbf{m}$ in the object space can have multiple projections. In other words, the camera samples rays of the plenoptic function by having multiple projection centers. Thus, one wants to maximize the number of projections obtained from the projection model.

3.4.1 Set of Imaged Rays

A point in the object space projects into a line (projection line) in the ray-spaces (i,k) and (j,l) (blue line in Figure 3.3). The projection defined in (3.14) has 4 unknowns (i,j,k,l) and 2 equations, which is not enough to define the rays $\mathbf{\Phi}$ on the sensor plane without any knowledge of the LF. Thus, one assumes that the LF size is known. In a real camera one has a finite LF size that implies a finite number of rays $\mathbf{\Phi}$ obtained for the projection of a point $\mathbf{m}$.

Using the LF size and considering the discretization that occurs at the image sensor, one can assume integer values for the microlenses and determine the corresponding pixels. Nonetheless, according to the slope of the projection lines one can skip some projections since the coordinates (k,l) are restricted to be integers (red pixels in Figure 3.3.a). The same occurs if one assumes integer values for the pixels and determine the



(a) (b)

Figure 3.3: Rasterization method used to obtain the projections of a point $\mathbf{m}$ for the (i, k) coordinates for different slopes of the projection line $i = \mathrm{f}^{-1}(k; \mathbf{m}, \mathbf{H})$. The red pixels correspond to the projections skipped by assuming integer values for the microlenses k.

corresponding microlenses. Since it is desired to maximize the number of projections, one should evaluate the slope of the projection lines to determine which coordinates are more discriminative, the pixels or the microlenses.

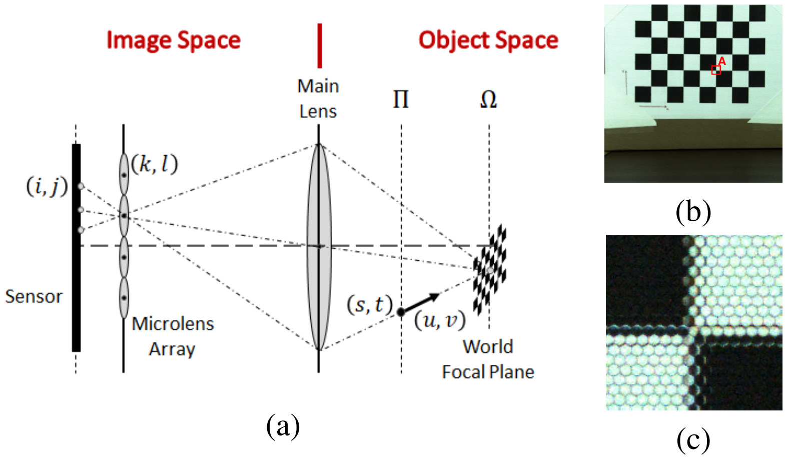
Considering the affine mappings $k = \mathrm{f}(i; \mathbf{m}, \mathbf{H}) = m_k \; i + b_k$ and $l = \mathrm{g}(j; \mathbf{m}, \mathbf{H}) = m_l \; j + b_l$, the slope of the projection lines $m_{(\cdot)}$ corresponds to the disparity between Viewpoint Images (VIs), and its inverse corresponds to the disparity between MIs. Slope $m_{(\cdot)}$ can be identified in (3.14) as the factor multiplying i or j, namely

$$m_k = -\frac{h_{si} + z\, h_{ui}}{h_{sk} + z\, h_{uk}} \quad , \qquad m_l = -\frac{h_{tj} + z\, h_{vj}}{h_{tl} + z\, h_{vl}} \quad . \tag{3.15}$$

Notice that the slope is constant for points at the same depth. $b_{(\cdot)}$ is the k- or l-intercept. To simplify, in the following, consider that the optical setup is point symmetric, *i.e.* the setup has square pixels and equally spaced microlenses in both vertical and horizontal directions. This implies that $\mathrm{f}(i; \mathbf{m}, \mathbf{H}) \equiv \mathrm{g}(j; \mathbf{m}, \mathbf{H})$. Hence, if $\left|m_{(\cdot)}\right| \leq 1$, the pixels are more discriminative (Figure 3.3.a) and the microlens are given by the set $\mathcal{P}_{kl}$

$$\left\{[i,j,k,l]^T : \ k = \mathrm{f}\left(i; \mathbf{m}, \mathbf{H}\right), \ l = \mathrm{g}\left(j; \mathbf{m}, \mathbf{H}\right), \ i \in \mathbb{N}_i, \ j \in \mathbb{N}_j \ \right\} \tag{3.16}$$

where $\mathbb{N}_i = \{0, \ldots, N_i - 1\} \subset \mathbb{N}_0^+$, $\mathbb{N}_j = \left\{0, \ldots, N_j - 1\right\} \subset \mathbb{N}_0^+$, and N_i and N_j correspond to the number of pixels of the sensor in each of the dimensions i and j. This is the case where a point $\mathbf{m}$ projects to more than one pixel within each microlens. This occurs, for example, in an SPC, for points in the object space near the focal plane or in focus by the main lens (Figure 3.4).



(a)

(b)

(c)

Figure 3.4: Calibration grid placed on the main lens focal plane Ω for an SPC **(a)**. **(b)** shows the raw image of the calibration grid and **(c)** exhibits the details of the microlenses in red box A. Notice that the microlenses do not define a sharp corner.

On the other hand, if $\left|m_{(\cdot)}\right| > 1$, the microlenses are more discriminative (Figure 3.3.b) and the pixels are given by the set $\mathcal{P}_{ij}$

$$\left\{[i,j,k,l]^T : \ i = \mathrm{f}^{-1}\left(k; \mathbf{m}, \mathbf{H}\right), \ j = \mathrm{g}^{-1}\left(l; \mathbf{m}, \mathbf{H}\right), \ k \in \mathbb{N}_k, \ l \in \mathbb{N}_l \right\} \tag{3.17}$$

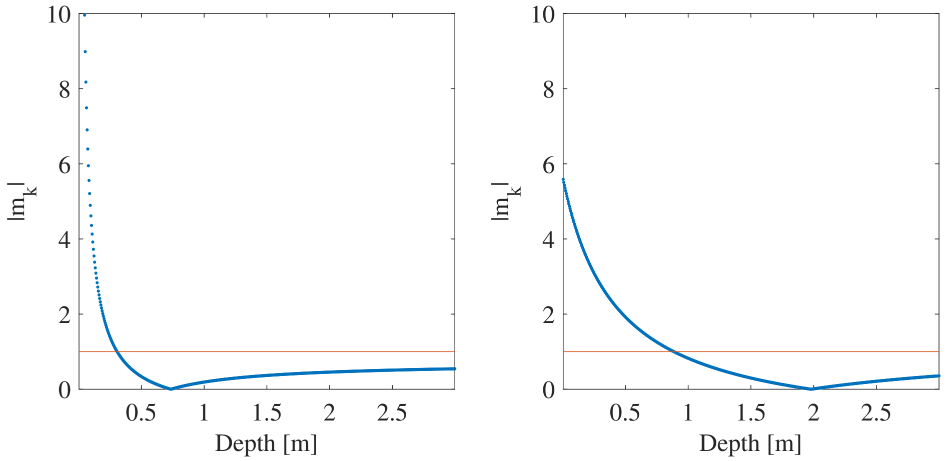
where $\mathbb{N}_k = \{0, \ldots, N_k - 1\} \subset \mathbb{N}_0^+$, $\mathbb{N}_l = \{0, \ldots, N_l - 1\} \subset \mathbb{N}_0^+$, and N_k and N_l correspond to the number of microlenses in each of the dimensions k and l. Since the camera might deviate from this point symmetric behavior, one should consider a correction using a mixture of the sets $\mathcal{P}_{ij}$ and $\mathcal{P}_{kl}$. For example, by considering $k = \mathrm{f}\left(i; \mathbf{m}, \mathbf{H}\right)$ and

$j = \mathrm{g}^{-1}(l; \mathbf{m}, \mathbf{H})$. The sets $\mathcal{P}_{ij}$ and $\mathcal{P}_{kl}$ describe a rasterization method for representing the lines defined in (3.14) for each of the coordinate pairs (i, k) and (j, l) in terms of discretized indices for pixels and microlenses. This process allows to implicitly overcome the limitations, detailed in Section 3.4.2, of the projection (3.14).

The conditions described previously to apply each of the projection sets do not directly relate with the depth of a point in the object space. Redefining the conditions relatively to the depth z of a point leads to projection rays defined by the projection set $\mathcal{P}_{ij}$ (3.17) whenever $z \in \chi$ where $\chi = \left] -\frac{h_{si} + h_{sk}}{h_{uk} + h_{ui}}, \frac{h_{si} - h_{sk}}{h_{uk} - h_{ui}} \right[$. The projection set $\mathcal{P}_{kl}$ (3.16) is used whenever $z \notin \chi$.

The depth limits of the set χ are not easily interpretable expanding the entries of the LFIM with the parameters introduced in Section 3.5. Thus, in Figure 3.5, one relates the projection sets with the depth of a point z for the publicly available Datasets D and F of Monteiro *et al.* [107]. The depth of the point in the world coordinate system is defined as the distance to the encasing of the camera. This figure depicts that the projection set defined by $\mathcal{P}_{kl}$ is used for points farther from the camera while the projection set $\mathcal{P}_{ij}$ is used for points near the camera. Notice that the projection set $\mathcal{P}_{kl}$ is applied whenever the blue line is below the red line in Figure 3.5.

In summary, the complete ray-based projection model comprises the two sets, $\mathcal{P}_{ij}$ (3.17) and $\mathcal{P}_{kl}$ (3.16), nonetheless, for most depth values the projection rays are obtained using the set $\mathcal{P}_{kl}$. The set $\mathcal{P}_{ij}$ is only used for points near the camera. The ray-based projection model can be defined using Algorithm 1. For simplicity, is presented the algorithm assuming that the optical system is point symmetric. C corresponds to the number of projection rays obtained.



(a) Dataset D

(b) Dataset F

Figure 3.5: Evolution of slope $|m_k|$ (blue dots) with the depth of the points in the object space for Datasets D **(a)** and F **(b)**. The red line $|m_k| = 1$ defines the projection set that will be used to obtain the projection rays. For points below the red line, the set $\mathcal{P}_{kl}$ is used. For points above the red line, the set $\mathcal{P}_{ij}$ is used.

```
Algorithm 1: Project scene point m
Input  : Scene point: $\mathbf{m} = [x,\ y,\ z]^T$
         Parameters: $\mathbf{H},\ N_i,\ N_j,\ N_k,\ N_l$
Output: Projection Rays: $\{\mathbf{\Phi}_1, \ldots, \mathbf{\Phi}_C\}$
Compute the slope $m_k$ from equation (3.15)
if $|m_k| \leq 1$ then
    Rasterize $\mathbf{\Phi}_n = (i,\ j,\ k,\ l)$ according to set $\mathcal{P}_{kl}$ (3.16)
else
    Rasterize $\mathbf{\Phi}_n = (i,\ j,\ k,\ l)$ according to set $\mathcal{P}_{ij}$ (3.17)
end
```

3.4.2 Analysis of Singularities

The projection (3.14) has singularities. These singularities imply that some points in the object space have undefined projection rays (unobserved in the image). More precisely, $i = \mathrm{f}^{-1}(k; \mathbf{m}, \mathbf{H})$ or $k = \mathrm{f}(i; \mathbf{m}, \mathbf{H})$ are infinite for some depth values z, continuing with the point symmetric assumption.

The depth values for which the singularities occur are identified by $z_s^1 = -h_{si}/h_{ui}$ and $z_s^2 = -h_{sk}/h_{uk}$. Extending the definition of the entries h_{si}, h_{ui}, h_{sk}, and h_{uk} to consider the parameters presented in Section 3.5 for defining the LFIM, the singularities occur at $z_s^1 = \frac{d_M f_M}{d_M - f_M}$, and

$z_s^2 = \frac{d_M \, f_M \, (f_i - N \, f_k) + f_i \, d_\mu \, f_M}{(d_M - f_M) \, (f_i - N \, f_k) + f_i \, d_\mu}$. Considering that $N = N_i = N_j$ is the number of pixels in one dimension for the MI, d_M is the distance between the microlens plane and the main lens, d_μ is the distance between the sensor and the microlens array, and f_M is the focal length of the main lens. f_i and f_k are the spatial and directional sampling frequencies.

Looking more deeply into the singularities z_s^1 and z_s^2, it is possible to see that z_s^1 corresponds to points that lie on the focal plane of the main lens. This can be derived from the thin lens equation for the main lens and remembering that the LFIM $\mathbf{H}$ propagates the origin of a ray to a plane that corresponds to the main lens plane ($d = 0$). The depth of the singularity z_s^1 corresponds to the plane containing the projection centers of the microlens cameras (Section 5.1.2). This singularity occurs when the affine mapping $\mathrm{f}^{-1}\left(k; \mathbf{m}, \mathbf{H}\right)$ is applied. Implicitly, the singularity implies that the slope m_k^{-1} is undefined. Thus, in Section 3.4.1, the set $\mathcal{P}_{kl}$ (3.16) allows to overcome this limitation.

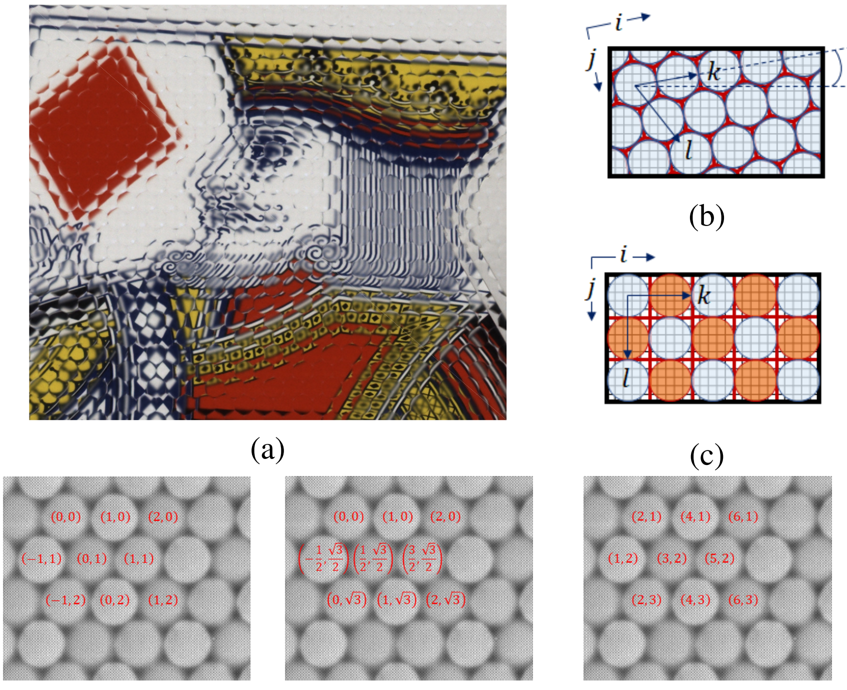
On the other hand, the singularity z_s^2 corresponds to the depth of the plane containing the projection centers of the viewpoint cameras (Section 4.1.2). The depth of the singularity is defined by the optical setup of the plenoptic camera and depends on several parameters including the sampling frequencies (see inline equation for z_s^2). This singularity occurs when one applies the affine mapping $\mathrm{f}\left(i; \mathbf{m}, \mathbf{H}\right)$. Implicitly, the singularity implies that the slope m_k is undefined. However, when this situation occurs, one uses the set $\mathcal{P}_{ij}$ (3.17) which avoids this limitation to occur in the projection model defined.

From these analysis, it is possible to see that, contrarily to a pinhole camera, a plenoptic camera can have projections even for points in the object space that are at the depths of the singularities z_s^1 and z_s^2.

3.5 Generalized LFIM

The LFIM appears in the literature to represent the SPC [41] and FPC [150]. The structure of the LFIM (3.2) used to represent both optical setups is the same. In this section, one shows the origin of the LFIM and the underlying assumptions for having the same LFIM structure for the SPCs and FPCs.

The LFIM proposed by Dansereau *et al.* [41] (3.2) describes a virtual plenoptic camera whose microlenses define a rectangular tiling. The corresponding LF is obtained after a decoding process that comprises segmentation of the MIs, alignment of the image sensor relatively to the microlens array, and hexagonal sampling correction (Figure 3.6.c) to transform the actual 2D raw image captured by a plenoptic camera. However, a plenoptic camera has a microlens array with hexagonal tiling that is not aligned with the image sensor (Figure 3.6.b). Thus, the camera model for a plenoptic camera should include the decoding transformations considered for the virtual plenoptic camera.



(a)

(b)

(c)

(d) Axial [67] (e) Zhang *et al.* [150] (f) Monteiro *et al.* [108]

Figure 3.6: Real and virtual microlens array structure of a plenoptic camera. The real microlens array **(b)** defines an hexagonal tiling that is not aligned with the image sensor by an angle θ and can be represented using an axial coordinate system **(d-e)** or a cartesian coordinate system **(f)**. The microlens array with the hexagonal structure can be identified in the raw image of an MPC [5] **(a)**. The virtual microlens array **(c)** created by Dansereau *et al.* [41] defines a rectangular tiling that is aligned with the image sensor. The virtual microlens array is obtained after a decoding process whose rays of the missing microlenses (in orange) are estimated by interpolation.

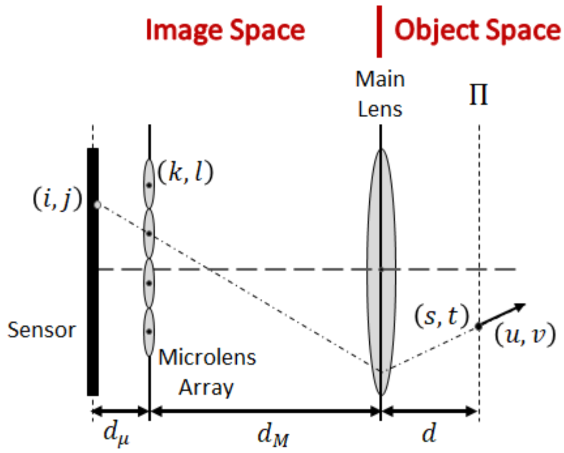


Figure 3.7: Geometry of a plenoptic camera considering the microlenses as pinholes and the main lens as a thin lens.

The LFIM that describes a plenoptic camera and maps the rays in the image space to the rays in the object space (Figure 3.7) is obtained by applying a series of seven transformations

$$\mathbf{H} = \mathbf{H}^{M\to\Pi}\mathbf{H}^{M}\mathbf{H}^{S\to M}\mathbf{H}^{\phi}_{m}\mathbf{H}^{m}_{a}\mathbf{H}^{a}_{a\mu}\mathbf{H}^{a\mu}_{r} \quad , \tag{3.18}$$

resulting in a 5×5 matrix $\mathbf{H}$ with 20 non-zero entries

$$\mathbf{H} = \begin{bmatrix} h_{si} & h_{sj} & h_{sk} & h_{sl} & h_{s} \\ h_{ti} & h_{tj} & h_{tk} & h_{tl} & h_{t} \\ h_{ui} & h_{uj} & h_{uk} & h_{ul} & h_{u} \\ h_{vi} & h_{vj} & h_{vk} & h_{vl} & h_{v} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad . \tag{3.19}$$

The hexagonal grid of microlenses can be represented by indices (k, l) using an axial coordinate system [67] whose basis differ from the standard cartesian coordinate system (Figure 3.6). The transformation between the two different coordinate systems makes the ray coordinates dependent, and therefore the LF coordinates should be analyzed simultaneously. Thus, starting from a ray $\tilde{\mathbf{\Phi}} = [i, j, k, l, 1]^T$ in homogeneous coordinates, the transformation

$$\mathbf{H}_r^{a\mu} = \underbrace{\begin{bmatrix} 1 & 0 & \multicolumn{2}{c}{\multirow{2}{*}{$\mathbf{N}_\mu$}} & -n_i \\ 0 & 1 & & & -n_j \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{H}_n} \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \multicolumn{2}{c}{\multirow{2}{*}{$\mathbf{S}_\mu$}} & 0 \\ 0 & 0 & & & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{H}_s} \tag{3.20}$$

converts the LF coordinates using an axial coordinate system to pixel coordinates using a cartesian coordinate system where $\left(n_i, n_j\right)$ defines a translational pixel offset. This mapping can describe the hexagonal sampling of the microlenses in several ways. For example, considering the width (N_i) and height (N_j) of the microlenses in pixels, the microlens sampling is defined as

$$\mathbf{N}_\mu^A = \begin{bmatrix} N_i & 0 \\ 0 & N_j \end{bmatrix} \quad , \quad \mathbf{S}_\mu^A = \begin{bmatrix} 1 & -\frac{1}{2} \\ 0 & \frac{3}{4} \end{bmatrix} \quad . \tag{3.21}$$

Alternatively, one can consider the distance R between the center of the hexagon that includes the MI and the hexagon corners, or considering the horizontal (d_h) and vertical (d_v) distances between consecutive microlenses centers. In these cases, the microlens sampling would be defined as

$$\begin{aligned} \mathbf{N}_\mu^B &= \begin{bmatrix} \sqrt{3}R & 0 \\ 0 & 2R \end{bmatrix} \quad , \quad \mathbf{S}_\mu^B = \begin{bmatrix} \sqrt{3} & -\frac{\sqrt{3}}{2} \\ 0 & \frac{3}{2} \end{bmatrix} \quad \text{or} \\ \mathbf{N}_\mu^C &= \begin{bmatrix} d_h & 0 \\ 0 & \frac{4}{3}d_v \end{bmatrix} \quad , \quad \mathbf{S}_\mu^C = \begin{bmatrix} 1 & -\frac{1}{2} \\ 0 & 1 \end{bmatrix} \quad , \end{aligned} \tag{3.22}$$

respectively. Additionally, the misalignment of the microlens array relatively to the image sensor introduces more dependencies among the coordinates of the ray in image space. This misalignment is described by the mapping

$$\mathbf{H}^{a}_{a\mu} = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 & -c_i \\ -\sin\theta & \cos\theta & 0 & 0 & -c_j \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.23}$$

that encodes a rotation θ and a translation of $\left(c_i, c_j\right)$ pixels of the microlens array relatively to the image sensor. This transformation obtains the 2D image sensor coordinates (pixels in the 2D raw image).

In the remaining transformations, the dimensions corresponding to the coordinates (i, k) and (j, l) are independent, and therefore the series of transformations can be analyzed separately for each pair of coordinates without loss of generality [41]. Thus, starting with the homogeneous coordinates $[i', k', 1]^T$, the transformation

$$\mathbf{H}^{m}_{a} = \begin{bmatrix} \frac{1}{f_i} & 0 & -\frac{o_i}{f_i} \\ 0 & \frac{1}{f_k} & -\frac{o_k}{f_k} \\ 0 & 0 & 1 \end{bmatrix} \tag{3.24}$$

converts the 2D image sensor coordinates and microlenses coordinates to metric coordinates by assuming that there are $f_{(\cdot)}$ samples per meter and an offset $o_{(\cdot)}$. This allows to define the 4D ray using two points defined in two planes, the image sensor and the microlens array. On the other hand, the mapping

$$\mathbf{H}^{\phi}_{m} = \begin{bmatrix} 1 & 0 & 0 \\ -\frac{1}{d_\mu} & \frac{1}{d_\mu} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.25}$$

allows to change the two-plane parameterization of the ray to a point and a direction defined in the image sensor plane using the distance between the image sensor and the microlens array (d_μ). This parameterization allows to use ray transfer matrices to propagate the ray to an arbitrary plane. Namely,

$$\mathbf{H}^{S\to M} = \begin{bmatrix} 1 & d_\mu + d_M & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.26}$$

propagates the ray in free space from the image sensor to the main lens and defines the position of the ray in the main lens. d_M is the distance between the microlens array and the main lens. Additionally, the mapping

$$\mathbf{H}^{M} = \begin{bmatrix} 1 & 0 & 0 \\ -\frac{1}{f_M} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.27}$$

describes the refraction that occurs at the main lens with focal length f_M. This allows to obtain the direction (u, v) in the object space without being modified by the optics of the plenoptic camera. Finally, the transformation

$$\mathbf{H}^{M\to\Pi} = \begin{bmatrix} 1 & d & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.28}$$

defines the origin of the ray in the object space at a point (s, t) in an arbitrary plane Π at a distance d from the main lens. The seven transformations allow to parameterize the ray in metric units by a point in plane Π and a direction (Figure 3.7).

The LFIM (3.19) is more complex than the LFIM proposed by Dansereau *et al.* [41] (3.2). However, it is possible to simplify (3.19) and reduce the number of non-zero entries. Normally, the misalignment between the microlens array and the image sensor is small, and in recent plenoptic cameras like Raytrix can be ignored. In the virtual plenoptic camera, the misalignment is also not considered in the camera model since it is corrected in the decoding process. Thus, one can consider that $\mathbf{H}^{a}_{a\mu} = \mathbf{I}_{5\times 5}$ where $\mathbf{I}_{5\times 5}$ is a 5×5 identity matrix which simplifies the LFIM $\mathbf{H}$ (3.19) to 14 non-zero entries

$$\mathbf{H} = \begin{bmatrix} h_{si} & 0 & h_{sk} & h_{sl} & h_s \\ 0 & h_{tj} & 0 & h_{tl} & h_t \\ h_{ui} & 0 & h_{uk} & h_{ul} & h_u \\ 0 & h_{vj} & 0 & h_{vl} & h_v \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} . \tag{3.29}$$

In order to further reduce the number of non-zero entries, one should choose an appropriate coordinate system for the microlens coordinates (k, l). Namely, one can incorporate the axial coordinate system basis in the microlenses coordinates as in Zhang *et al.* [150] (Figure 3.6.e) considering the microlens coordinates defined as $[k_z, l_z]^T = \mathbf{S}_\mu \, [k, l]^T$ with $\mathbf{s}_\mu = \begin{bmatrix} \sqrt{3} & -\frac{\sqrt{3}}{2} \\ 0 & \frac{3}{2} \end{bmatrix}$. In the series of transformations that lead to the matrix $\mathbf{H}$ (3.18), this is equivalent to separate the contributions of the matrices $\mathbf{H}_n$ and $\mathbf{H}_s$ in the mapping $\mathbf{H}^{a\mu}_{r}$ (3.20). More specifically, the matrix $\mathbf{H}_n$ will be used to define the LFIM $\mathbf{H}_z = \mathbf{H}^{M\to\Pi}\mathbf{H}^{M}\mathbf{H}^{S\to M}\mathbf{H}^{\phi}_{m}\mathbf{H}^{m}_{a}\mathbf{H}^{a}_{a\mu}\mathbf{H}_n$ while the matrix $\mathbf{H}_s$ will be used to define the ray coordinates in the image space $\mathbf{\Phi}_z = \mathbf{H}_s\mathbf{\Phi}$. However, this originates non-integer indices for the microlenses which might difficult the access to a particular microlens.

Alternatively, one can use a rectangular sampling basis without resorting to a decoding process (Figure 3.6.f) [108]. This allows to represent the hexagonal structure of the microlens centers in raw image coordinates (p, g) using integer (k, l) coordinates given by $[p, g]^T = \mathbf{N}_\mu \, \mathbf{S}_\mu \, [k, l]^T + [p_0, g_0]^T$ where $\mathbf{N}_\mu = \text{diag}(d_h, d_v)$, $\mathbf{S}_\mu = \mathrm{diag}(\frac{1}{2}, 1)$ and (p_0, g_0) correspond to the origin for the (k, l) coordinates in the raw image.

In the virtual plenoptic camera, the decoding process corrects the microlens hexagonal sampling by generating a rectangular tiling of the microlenses [41] which result in a sampling that is described by a rectangular basis $\mathbf{S}_\mu = \mathbf{I}_{2\times 2}$ where $\mathbf{I}_{2\times 2}$ is the 2×2 identity matrix and considering $\mathbf{N}_\mu = \text{diag}(N_i, N_j)$.

The misalignment simplification and the appropriate choice of the microlens coordinates allow to model a plenoptic camera (SPC and FPC) with a LFIM $\mathbf{H}$ with 12 non-zero entries (3.2), identical to the one described by Dansereau *et al.* [41].

3.6 Chapter Summary

In this chapter, was introduced the LFIM that maps rays in the image space $\mathbf{\Phi}$ to rays in the object space $\mathbf{\Psi}$. This mapping was used to describe a ray-based projection model for a plenoptic camera defined by the sets $\mathcal{P}_{ij}$ (3.17) and $\mathcal{P}_{kl}$ (3.16). This projection model is derived from (3.14) considering the affine mappings $\text{f}\left(i; \mathbf{m}, \mathbf{H}\right)$ and $\text{g}\left(j; \mathbf{m}, \mathbf{H}\right)$ and the goal of maximizing the number of projections.

Additionally, were explained the assumptions that allow to have an identical LFIM $\mathbf{H}$ with 12 non-zero entries (3.2), identical to the one described by Dansereau *et al.* [41], for SPCs and FPCs. More specifically, the same structure is obtained by the appropriate choice of the microlens coordinates and by assuming that the image sensor is aligned with the microlens array.

The ray-based projection model relies completely on the LFIM and on the ray definition. This approach does not try to associate a meaning to each of the LFIM parameters and does not explain the consequence of either fixing the coordinates (i, j) or (k, l) to obtain the projections of a point. Chapters 4 and 5 will deepen the understanding of the LFIM and of the plenoptic camera projection model by defining the projection models associated with the viewpoint and microlens cameras.

Chapter 4

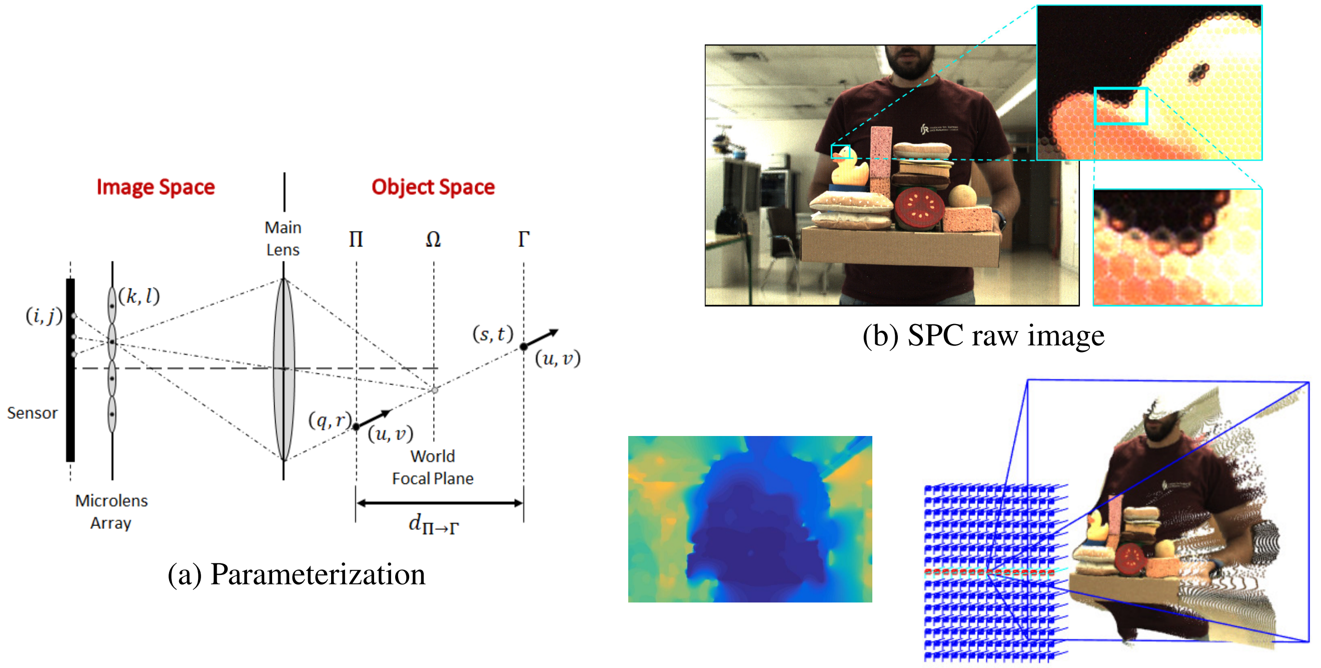
Standard Plenoptic Camera

The geometry model most used for Standard Plenoptic Cameras (SPCs) is the one proposed by Dansereau *et al.* [41] and described in Chapter 3. This model maps rays in the image space indexed by pixels and microlenses indices to rays in the object space defined in metric units. The concept of Viewpoint Image (VI) defined by Ng *et al.* [112] (Section 2.5.1), obtained by selecting the same pixel for each microlens, is normally used to represent the Lightfield (LF) obtained by an SPC and allows to conveniently view the SPC as a camera array (Figure 4.1.d).

This chapter starts from the model of Dansereau *et al.* [41] and has a derivation of the mapping between the Lightfield Intrinsic Matrix (LFIM) and the viewpoint camera array that allows to fully formalize the projection model for a viewpoint camera. Two calibration approaches are proposed for an SPC based on the geometry of the viewpoint array. The depth capabilities of an SPC are evaluated for the depth range between 0.05 and 2.00 m.

4.1 Viewpoint Camera Array

In this section, is shown that the LFIM [41] can represent an array of distinct coplanar and parallel cameras (Figure 4.1.d). The VI is obtained by selecting the same pixel (i, j) of each microlens (k, l). In this case, the coordinates (i, j) are the indices associated with each VI and the coordinates (k, l) encode the position of a pixel in the VI. Let us consider the projection matrix $\mathbf{P}^{ij}$ to describe a viewpoint camera parameterized



(a) Parameterization

(b) SPC raw image

(c) Reconstructed depth map

(d) 3D reconstruction and camera array (centers spaced $50\times$)

Figure 4.1: LF parameterization, scene reconstruction and viewpoint camera array. **(a)** The LF in the image space is parameterized using pixels and microlenses indices while the LF in the object space is parameterized using a point and a direction. **(b)** Image captured on the sensor of an SPC. **(c)** depicts the depth map obtained using [95]. **(d)** Viewpoint camera array obtained by calibration where the spacing among projection centers has been scaled 50 times to be perceptible on the 3D plot.

by the indices $(i, j) \in \mathbb{Z}^2$

$$\mathbf{P}^{ij} = \mathbf{K}^{ij} \begin{bmatrix} \mathbf{I}_{3\times 3} & \mathbf{t}^{ij} \end{bmatrix} {}^{c}\mathbf{T}_{w} \tag{4.1}$$

where $\mathbf{K}^{ij}$ denotes the intrinsic matrix, $\mathbf{I}_{3\times 3}$ is a 3×3 identity matrix, $\mathbf{t}^{ij}$ is the projection center and ${}^{c}\mathbf{T}_{w} = \begin{bmatrix} {}^{c}\mathbf{R}_{w} & {}^{c}\mathbf{t}_{w} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}$ defines the rigid body transformation between the world and camera coordinate systems with rotation ${}^{c}\mathbf{R}_{w} \in SO(3)$ and translation ${}^{c}\mathbf{t}_{w} \in \mathbb{R}^3$, and $\mathbf{0}_{1\times 3}$ corresponds to the 1×3 null matrix.

Note that while ${}^{c}\mathbf{T}_{w}$ defines one coordinate system for all cameras, the intrinsic matrix and the projection center are different for each viewpoint camera (i, j) in the array. In the following, let the camera model (4.1) for the viewpoint cameras in the array take into account that the principal point and the projection center are different for each camera while the scale factor remains the same:

$$\mathbf{K}^{ij} = \begin{bmatrix} k_u & 0 & u_0 + i\,\Delta u_0 \\ 0 & k_v & v_0 + j\,\Delta v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{t}^{ij} = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} + \begin{bmatrix} i\,\Delta x_0 \\ j\,\Delta y_0 \\ 0 \end{bmatrix} \tag{4.2}$$

where the scalars k_u and k_v denote focal lengths and conversion from metric units to pixels (denominated as scale factors in the remainder of the thesis). The vector $[u_0, v_0]^T$ defines the principal point for the viewpoint camera $(i, j) = (0, 0)$, and the vectors $[\Delta u_0, \Delta v_0]^T$ and $[\Delta x_0, \Delta y_0, 0]^T$ denote principal point shift and baseline between consecutive cameras in the array, respectively. The vector $[x_0, y_0, z_0]^T$ defines the location of the camera array relatively to the camera coordinate system origin. This allows to represent the array of cameras using a maximum of 11 parameters.

4.1.1 Projection Model

Considering the projection matrix (4.1), one can obtain the multiple projections for a point $\mathbf{m} = [x, y, z]^T$ in the object space considering the available camera indices. More specifically, let us define the projection matrix of a particular camera (i, j) as

$$\mathbf{P}^{ij} = \mathbf{P}^0 + i\,\mathbf{\Delta P}^i + j\,\mathbf{\Delta P}^j \tag{4.3}$$

with

$$\begin{aligned}
\mathbf{P}^0 &= \begin{bmatrix} k_u & 0 & u_0 \\ 0 & k_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} & x_0 \\ \mathbf{I}_{3\times3} & y_0 \\ & z_0 \end{bmatrix} {}^c\mathbf{T}_w \quad , \\
\boldsymbol{\Delta}\mathbf{P}^i &= \begin{bmatrix} \mathbf{0}_{3\times2} & \begin{matrix} \Delta u_0 & k_u\,\Delta x_0 + \Delta u_0\, z_0 \\ \multicolumn{2}{c}{\mathbf{0}_{2\times2}} \end{matrix} \end{bmatrix} {}^c\mathbf{T}_w \quad \text{and} \\
\boldsymbol{\Delta}\mathbf{P}^j &= \begin{bmatrix} \mathbf{0}_{3\times2} & \begin{matrix} \multicolumn{2}{c}{\mathbf{0}_{1\times2}} \\ \Delta v_0 & k_v\,\Delta y_0 + \Delta v_0\, z_0 \\ \multicolumn{2}{c}{\mathbf{0}_{1\times2}} \end{matrix} \end{bmatrix} {}^c\mathbf{T}_w \quad .
\end{aligned} \tag{4.4}$$

In these equations, $\mathbf{P}^0$ defines a projection matrix that does not have a dependency on the camera indices, *i.e.* the projection matrix for the viewpoint camera $(i, j) = (0, 0)$. The incremental matrices $\boldsymbol{\Delta}\mathbf{P}^i$ and $\boldsymbol{\Delta}\mathbf{P}^j$ give the contribution of each camera index for defining the projection matrix $\mathbf{P}^{ij}$ for an arbitrary viewpoint camera (i, j) and $\mathbf{0}_{n\times m}$ corresponds to the $n \times m$ null matrix.

Using (4.3), the projection of a point $\mathbf{m}$ to a point in the image plane $\tilde{\mathbf{q}} = [k, l, 1]^T$ of a particular camera (i, j) is given by

$$\tilde{\mathbf{q}} \sim \underbrace{\begin{bmatrix} \mathbf{I}_{3\times3} & i\,\mathbf{I}_{3\times3} & j\,\mathbf{I}_{3\times3} \end{bmatrix}}_{\mathbf{M}} \begin{bmatrix} \mathbf{P}^0 \\ \boldsymbol{\Delta}\mathbf{P}^i \\ \boldsymbol{\Delta}\mathbf{P}^j \end{bmatrix} \tilde{\mathbf{m}} \quad , \tag{4.5}$$

where the symbol $\sim$ denotes equal up to a scale factor. The matrix $\mathbf{M}$ provides an easy way to add the several camera indices available for a plenoptic camera and in this way get the multiple projections for a point $\mathbf{m}$ in the object space.

The projection (4.5) using the viewpoint coordinates (i, j) is equivalent to the projection set $\mathcal{P}_{kl}$ defined in Section 3.4.1.

4.1.2 Mapping from LFIM to Viewpoint Projection Matrices

In order to obtain the mapping from the LFIM to the camera model (4.1) let us first define the projection centers of the viewpoint cameras, and then define the projection equation considering the LFIM $\mathbf{H}$ and (i, j) as parameters.

Viewpoint Projection Centers. Let us consider a LF in the object space $L_{\Pi}(q, r, u, v)$ acquired by a plenoptic camera with the plane Ω in focus (Figure 4.1.a). $L_{\Pi}(q, r, u, v)$ is a set of rays, where each ray $\tilde{\mathbf{\Psi}}_{\Pi} = [q, r, u, v, 1]^T$ is parameterized using a point (q, r) on a plane Π and a direction (u, v) defined in metric units [107]. This LF is mapped to the LF in the image space $L(i, j, k, l)$ by the LFIM $\mathbf{H}_{\Pi}$:

$$\tilde{\mathbf{\Psi}}_{\Pi} = \mathbf{H}_{\Pi}\, \tilde{\mathbf{\Phi}} \quad , \tag{4.6}$$

where $\tilde{\mathbf{\Phi}} = [i, j, k, l, 1]^T$ corresponds to a ray that is parameterized by pixels (i, j) and microlenses (k, l) indices and

$$\mathbf{H}_{\Pi} = \begin{bmatrix} h_{qi} & 0 & h_{qk} & 0 & h_q \\ 0 & h_{rj} & 0 & h_{rl} & h_r \\ h_{ui} & 0 & h_{uk} & 0 & h_u \\ 0 & h_{vj} & 0 & h_{vl} & h_v \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad . \tag{4.7}$$

This mapping allows writing the positions (q, r) and the directions (u, v) as affine mappings on the pixels (i, j) and microlenses (k, l) indices.

For a viewpoint or sub-aperture camera, the pixel coordinates (i, j) are fixed and are considered as parameters. Hence, for a viewpoint camera, the positions (q, r) and the directions (u, v) are affine mappings only on the microlens coordinates (k, l), namely

$$\begin{cases} q\,(k;\ i, \mathbf{H}_\Pi) = h_{qk}\, k + h_{qi}\, i + h_q \\ r\,(l;\ j, \mathbf{H}_\Pi) = h_{rl}\, l + h_{rj}\, j + h_r \\ u\,(k;\ i, \mathbf{H}_\Pi) = h_{uk}\, k + h_{ui}\, i + h_u \\ v\,(l;\ j, \mathbf{H}_\Pi) = h_{vl}\, l + h_{vj}\, j + h_v \end{cases} \tag{4.8}$$

where the LFIM $\mathbf{H}_\Pi$ is also considered as a parameter. To simplify the notation, the parameters $(i, j, \mathbf{H}_\Pi)$ will not be included in the following expressions.

A ray captured by a plenoptic camera and parameterized by (i, j, k, l) intersects the plane Π at point $\mathbf{p}\,(k, l) = [q(k),\ r(l),\ 0]^T$ with a direction $\mathbf{n}\,(k, l) = [u(k),\ v(l),\ 1]^T$. This allows to define an arbitrary point $\mathbf{c}\,(k, l, \lambda) = [x, y, z]^T$ along the ray [58] as

$$\mathbf{c}\,(k, l, \lambda) = \mathbf{p}\,(k, l) + \lambda\, \mathbf{n}\,(k, l)\ ,\ \lambda \in \mathbb{R} \quad . \tag{4.9}$$

Note that by sweeping the range of (k, l) in (4.9) with $\lambda = 0$, one samples an area of the plane Π through which pass all the viewpoint imaging rays. In addition, by sweeping (i, j), one obtains all the viewpoints, and therefore all rays that can be imaged by the plenoptic camera. Finally, sweeping λ, allows representing all world points within the Field of View (FOV) of the plenoptic camera.

The location of the projection centers of an optical setup is defined by its caustic surface, which is the loci of singularities in the flux density [27, 58]. The convergence of the rays captured by a camera at a single point, *i.e.* a unique projection center, is considered a degenerate configuration of the caustic surface (point caustic) [58]. Although there are many techniques to derive the caustic surface, one will consider the Jacobian method [27].

The caustic surface is defined at the points in the object space where the ray to image mapping (4.9) is singular, *i.e.* the mapping from (k, l, λ) to (x, y, z) is singular. The singularities occur at the set of points where the Jacobian matrix of the transformation does not have full rank, *i.e.* points that make the determinant of the Jacobian vanish $\det\Big(\mathbf{J}\left(\mathbf{c}\left(k,l,\lambda\right)\right)\Big) = 0$. Solving the vanishing constraint one obtains two solutions for λ:

$$\lambda_1 = -\frac{h_{qk}}{h_{uk}} \quad \vee \quad \lambda_2 = -\frac{h_{rl}}{h_{vl}} \quad . \tag{4.10}$$

Replacing λ_1 or λ_2 in (4.9), one identifies the caustic profile for the viewpoint camera. The caustic profile of a single viewpoint consists of a line with (i) unique (x, z) and variable y components if $\lambda = \lambda_1$ or (ii) unique (y, z) and variable x components if $\lambda = \lambda_2$. In case $\lambda_1 \neq \lambda_2$ the viewpoint is a non-central camera. The viewpoint camera corresponds to a central camera, *i.e.* a camera with a unique projection center, if and only if $\lambda_1 = \lambda_2$ which imply the model parameters relation

$$\frac{h_{qk}}{h_{uk}} = \frac{h_{rl}}{h_{vl}} \quad . \tag{4.11}$$

Assuming this constraint and replacing λ in (4.9), expanded by the expressions in (4.8), the location of the viewpoint projection center for a viewpoint camera (i, j) is given by

$$\mathbf{p}_c = \begin{bmatrix} h_q - \frac{h_{qk}}{h_{uk}} h_u + i\left(h_{qi} - \frac{h_{qk}}{h_{uk}} h_{ui}\right) \\ h_r - \frac{h_{rl}}{h_{vl}} h_v + j\left(h_{rj} - \frac{h_{rl}}{h_{vl}} h_{vj}\right) \\ -\frac{h_{qk}}{h_{uk}} \end{bmatrix} \quad . \tag{4.12}$$

Furthermore, considering all viewpoint cameras that can be defined, the LFIM represents a coplanar grid of equally spaced projection centers. Notice that the pixels (i, j) only affect the x- and y-components of the projection centers while the z-component of the projections centers is always the same.

LFIM Mapping. Considering that the rays of one viewpoint camera converge to a unique point (4.11), one may set constant the values (i, j)

and solve (3.11) relatively to (k, l). This gives an equation of a viewpoint pixel (k, l) imaging the 3D point (x, y, z) that can be rewritten as a pinhole model like (4.1) with the intrinsic matrix defined as

$$\mathbf{K}^{ij} = \begin{bmatrix} \frac{1}{h_{uk}} & 0 & -\frac{h_u}{h_{uk}} - i\,\frac{h_{ui}}{h_{uk}} \\ 0 & \frac{1}{h_{vl}} & -\frac{h_v}{h_{vl}} - j\,\frac{h_{vj}}{h_{vl}} \\ 0 & 0 & 1 \end{bmatrix} \quad , \tag{4.13}$$

and the projection center as $\mathbf{t}^{ij} = -\mathbf{p}_c$ (4.12). This allows to obtain the mappings to the representations in (4.2). Namely, comparing (4.13) with (4.2), one identifies a common component $[u_0, v_0]^T = -\left[h_u/h_{uk}, h_v/h_{vl}\right]^T$ and a differential (shift) component $[\Delta u_0, \Delta v_0]^T = -\left[h_{ui}/h_{uk}, h_{vj}/h_{vl}\right]^T$ on the principal point. The scale factors are defined as $k_u = 1/h_{uk}$ and $k_v = 1/h_{vl}$, and the baseline is defined as $[\Delta x_0, \Delta y_0, 0]^T = -\left[h_{qi} - h_{ui}\ h_{qk}/h_{uk}\right.$
The position of the viewpoint camera array origin relatively to the camera coordinate system origin is defined as $[x_0, y_0, z_0]^T = -\left[h_q - h_u\ h_{qk}/h_{uk},\ h_r\right.$

An example of the pinhole model parameters for a viewpoint camera array obtained from a calibrated Lytro Illum camera can be found in Table 4.6. This array is configured for a focused depth of about 1.09 meters and describes 15×15 (i, j) cameras whose VIs have 625×433 (k, l) pixels.

4.1.3 Properties of Viewpoint Projection Matrices

Considering equation (3.11), one can obtain the Epipolar Plane Image (EPI) geometry that relates the depth of a point with the disparity on the VIs $\left[\frac{\Delta k}{\Delta i}, \frac{\Delta l}{\Delta j}\right]^T$

$$\frac{\Delta k}{\Delta i} = -\frac{h_{qi} - \frac{h_{qk}}{h_{uk}} h_{ui}}{h_{uk}} \frac{1}{z + \frac{h_{qk}}{h_{uk}}} - \frac{h_{ui}}{h_{uk}} \text{ and } \frac{\Delta l}{\Delta j} = -\frac{h_{rj} - \frac{h_{rl}}{h_{vl}} h_{vj}}{h_{vl}} \frac{1}{z + \frac{h_{rl}}{h_{vl}}} - \frac{h_{vj}}{h_{vl}} \,. \tag{4.14}$$

The mapping (4.12) and (4.13) allows to rewrite the EPI geometry defined in equation (4.14) as

$$\frac{\Delta k}{\Delta i} = k_u \frac{\Delta x_0}{z + z_0} + \Delta u_0 \quad \text{and} \quad \frac{\Delta l}{\Delta j} = k_v \frac{\Delta y_0}{z + z_0} + \Delta v_0 \ . \tag{4.15}$$

The EPI geometry shows that despite the parallel optical axis, the zero disparity plane, also known as the optical focal plane [112] of the main lens is at a finite depth due to the principal point shift (box B in Figure 4.2.b). Namely, the zero disparity plane corresponds to the plane Ω with $z_\Omega = -z_0 - k_u \frac{\Delta x_0}{\Delta u_0} = -z_0 - k_v \frac{\Delta y_0}{\Delta v_0}$. Contrarily, if one considers the principal point shift equal to zero, *i.e.* cameras with same principal point and therefore same intrinsic matrix $\mathbf{K}^{ij}$, one recovers the EPI geometry defined in [22] that defines points at infinity as the points of zero disparity [107] (Appendix B).

Looking at the EPIs obtained from a LF in Figure 4.2, one can see that the lines corresponding to different points in the object space have a range of positive and negative slopes. Namely, objects in the background (box A) have a negative slope while objects in the foreground (box C and D) have a positive slope. The disparity zero, in this scene, corresponds to the position of the person holding the objects (box B).

Notice also that the FOV is similar for all viewpoint cameras. Scene regions observed by the different viewpoint cameras change slightly for depths other than the zero disparity plane depth z_Ω (Figure 4.3.d). This is a consequence of the array of projection centers and array of principal points modeling viewpoint cameras. For the zero disparity plane depth $z_\Omega = -\frac{h_{qi}}{h_{ui}} = -\frac{h_{rj}}{h_{vj}}$, the influence of the different projection centers is cancelled by the principal point shift and the scene region observed is the same for all viewpoint cameras (Figure 4.3.c).

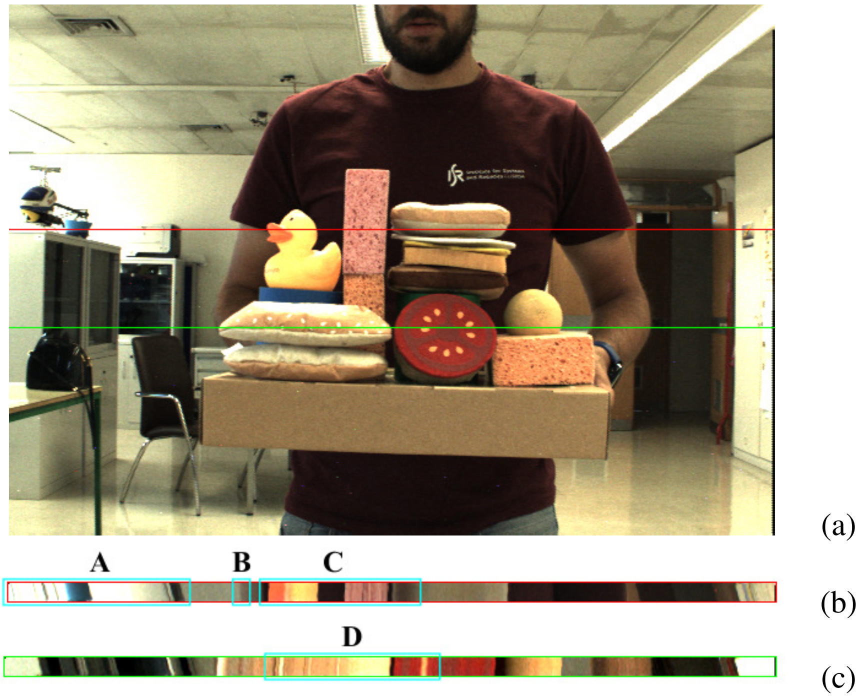
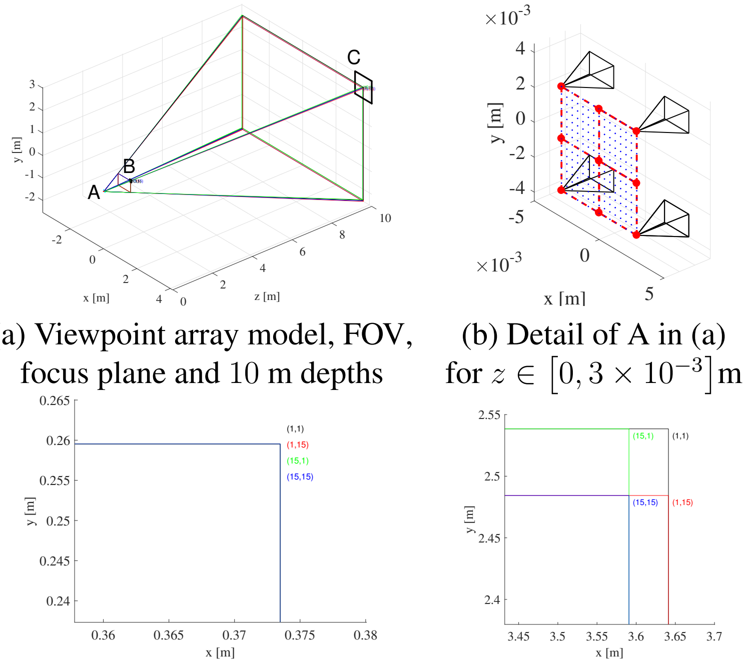


Figure 4.2: The viewpoint cameras identified in red in Figure 4.1.d are used to obtain EPIs from the LF at rows 185 (red) **(b)** and 265 (green) **(c)** on the central viewpoint **(a)**.

4.2 Reducing the Parameters of the LFIM

The LFIM has 12 non-zero entries (4.7) but some parameters can be avoided by considering them on the extrinsic parameters and choosing an appropriate camera coordinate system origin. Namely, choosing the camera coordinate system origin at the plane containing the viewpoint projection centers.

Considering the parameterization plane Π (Figure 4.1.a) for the origin of the different rays $\tilde{\mathbf{\Psi}}_\Pi = [q, r, u, v, 1]^T$ in the object space, an arbitrary point is defined as $[x, y, z]^T = [q, r, 0]^T + \lambda\ [u, v, 1]^T$, $\lambda \in \mathbb{R}$ [58]. The re-parameterization of the rays in the object space to the plane Γ (3.3) corresponds to a shift along the z-axis of the camera coordinate system, which results in $[x, y, z_\Gamma]^T = [s, t, 0]^T + \lambda\ [u, v, 1]^T$ where $s = q + u\, d_{\Pi\rightarrow\Gamma}$, $t = r + v\, d_{\Pi\rightarrow\Gamma}$, and $z_\Gamma = z - d_{\Pi\rightarrow\Gamma}$. Thus, the re-parameterization is redundant with the z-translation of the extrinsic parameters. Assuming that the plane Γ corresponds to the plane containing the viewpoint projection centers at $d_{\Pi\rightarrow\Gamma} = -h_{qk}/h_{uk}$, one obtains a LFIM $\mathbf{H}_\Gamma = \mathbf{D}_r\, \mathbf{H}_\Pi$



(a) Viewpoint array model, FOV, focus plane and 10 m depths

(b) Detail of A in (a) for $z \in \left[0, 3 \times 10^{-3}\right]$m

(c) Detail of B in (a)

(d) Detail of C in (a)

Figure 4.3: FOV of a Lytro Illum camera analyzed from the viewpoint array model. **(a)** back-projection pyramids of the four corner viewpoint cameras, $(i,j) = \left\{(1,1), (1,15), (15,1), (15,15)\right\}$, where A represents the array of projection centers, B is at the focus plane at depth z_Ω, and C is at depth $z = 10$ m. **(b)** zoom of A in (a), other viewpoint projection centers shown by red lines and blue dots. **(c)** zoom of black rectangle B in (a) showing the region observed at z_Ω is the same for all viewpoint cameras. **(d)** zoom of black rectangle C in (a) shows slight differences of regions observed by the different viewpoint cameras.

(3.4) with 10 non-zero entries

$$\mathbf{H}_\Gamma = \begin{bmatrix} h_{si} & 0 & 0 & 0 & h_s \\ 0 & h_{tj} & 0 & 0 & h_t \\ h_{ui} & 0 & h_{uk} & 0 & h_u \\ 0 & h_{vj} & 0 & h_{vl} & h_v \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} . \tag{4.16}$$

Furthermore, extending the definition of the point (s,t) to consider the rays $\tilde{\mathbf{\Phi}} = [i,j,k,l,1]^T$ in the image space (3.1) and redefining x and y as $x_\Gamma = x - h_s$ and $y_\Gamma = y - h_t$, one obtains $[x_\Gamma, y_\Gamma, z_\Gamma]^T = \left[h_{si}\, i, h_{tj}\, j, 0\right]^T + \lambda\, [u,v,1]^T$. Hence, the entries h_s and h_t are redundant with the (x,y)-translational components of the extrinsic parameters [41, 106]. Thus, removing the redundant entries, one obtains a LFIM $\mathbf{H}_\Gamma$

with 8 non-zero entries

$$\mathbf{H}_\Gamma = \begin{bmatrix} h_{si} & 0 & 0 & 0 & 0 \\ 0 & h_{tj} & 0 & 0 & 0 \\ h_{ui} & 0 & h_{uk} & 0 & h_u \\ 0 & h_{vj} & 0 & h_{vl} & h_v \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad . \tag{4.17}$$

Considering this representation for the LFIM, the viewpoint projection centers location (4.12) reduces to $\mathbf{p}_c = \left[i\ h_{si}, j\ h_{tj}, 0\right]^T$.

Considering that the SPC is described by the LFIM $\mathbf{H}_\Gamma$, one obtains a pinhole model with the intrinsic matrix (4.13) and with the projection center defined as

$$\mathbf{t}^{ij} = \begin{bmatrix} -i\ h_{si} \\ -j\ h_{tj} \\ 0 \end{bmatrix} \quad . \tag{4.18}$$

Comparing with (4.2), the baseline is defined as $[\Delta x_0, \Delta y_0, 0]^T = \left[-h_{si}, -h_{tj}, 0\right]$ and the vector $[x_0, y_0, z_0]^T = \mathbf{0}_{3\times 1}$ since the plane Γ coincides with the plane containing the viewpoint projection centers. This allows to represent the viewpoint camera array using 8 parameters.

The geometry described in this section allows to represent a coplanar camera array of distinct cameras that differ on their principal point. This representation also allows to represent a coplanar array of identical cameras by setting $\Delta u_0 = \Delta v_0 = 0$. Considering that the principal point shift is zero, the EPI geometry reduces to the one presented by Bolles *et al.* [22] (Appendix B).

4.3 Corner-based Calibration

The methods to calibrate SPCs normally consider corner features in VIs. The only work that considers other type of features is the one from Bok *et al.* [21] that uses lines in Microlens Images (MIs). However, these features are not detectable when the calibration pattern is placed near the mains lens world focal plane. In this region, the MIs consist of an image with very small deviations on the intensity values since these projections correspond to the same point in the scene [107] (Figure 3.4.c).

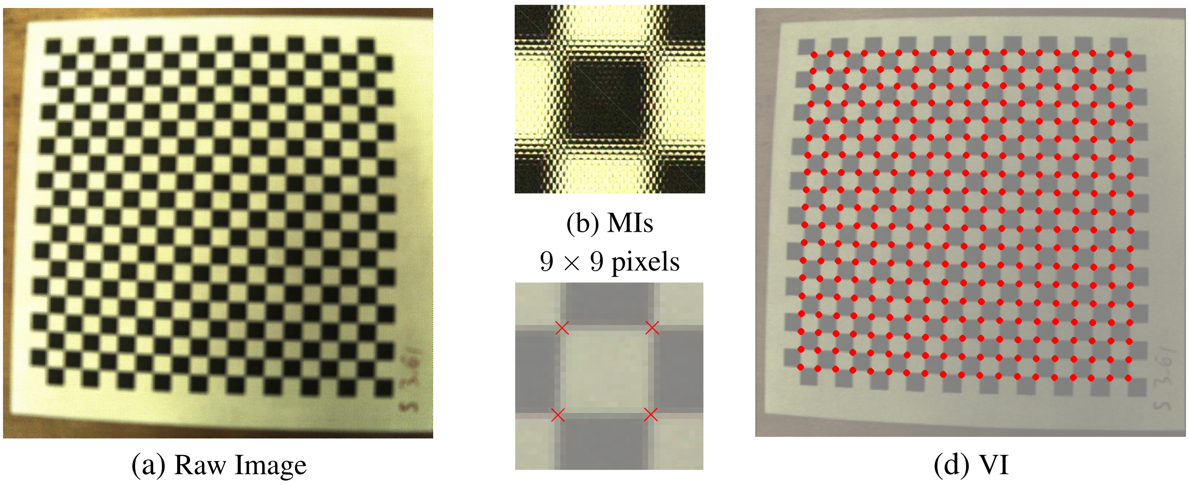
The VI corner-based methods that estimate the LFIM parameters differ mainly in the linear solution (Table 4.1). More specifically, the linear solution described in [41] estimates an homography for each viewpoint camera and pose of the calibration pattern. This solution estimates eight from the ten free intrinsic parameters, being the remaining two parameters estimated only in the nonlinear optimization. On the other hand, Zhang *et al.* [150] considers a reduced LFIM with 6 parameters similar to the one obtained for a coplanar array of identical cameras (Appendix B) [95]. This reduced representation is possible by including two extra-parameters on the radial distortion model of Brown [25] (Appendix A) which are only estimated in the nonlinear optimization.

Method	**Number Homographies**	**Number LFIM Parameters**	**Features**
Dansereau *et al.* [41]	$P \times C$	8 of 10	Corners in VIs
Bok *et al.* [21]	P	3 of 6	Lines in MIs
Zhang *et al.* [150]	P	6 of 8*	Corners in VIs
Monteiro *et al.* [104]	P	8 of 8	Corners in VIs

Table 4.1: State of the art comparison for SPC calibration procedures. P denotes the number of poses and C denotes the number of viewpoint cameras that can be obtained for an SPC. * Zhang *et al.* [150] considers a LFIM with 6 parameters being the other 2 included in the radial distortion model (Appendix A).

The full formalization of the viewpoint cameras projection model allows adapting methods from mainstream computer vision to plenoptic cameras. The linear solution of the calibration proposed in this section uses the mapping between the viewpoint projection model and the LFIM parameters to define a more efficient method to estimate the camera model parameters. Namely, one estimates a single generalized homography per pose of the calibration pattern, and extending techniques from pinhole camera calibration one recovers the eight free intrinsic parameters of the camera model. This is the first calibration procedure that allows to estimate the full LFIM in the linear solution.

In this section, the calibration proposed considers the corners of a planar calibration grid of known dimensions as features. In the following, one assumes that the corners in the world coordinate system have been matched with the imaged corners. Let us consider a 4D LF obtained from the raw image (Figure 4.1.b) after the decoding process [41, 44] (Section 2.5.1). An imaged corner is defined by a ray $\mathbf{\Phi} = [i, j, k, l]^T$ in the image space. The (k, l) coordinates correspond to the pixel coordinates of the detected corners on the VIs while the (i, j) coordinates correspond to the viewpoint coordinates. The (i, j) coordinates are integers and the (k, l) coordinates are real since generally a feature detector has sub-pixel accuracy (Figure 4.4.c).


(a) Raw Image
3280 × 3280 pixels

(b) MIs
9 × 9 pixels

(c) Detail of VI

(d) VI
383 × 381 pixels

Figure 4.4: **(a)** Debayered raw image from an SPC [41] with zoom **(b)** to show the effect of the microlens array. The features (k, l) obtained by the feature detector are shown in red for all calibration grid points **(d)**. The sub-pixel accuracy is depicted in **(c)**. The contrast is reduced for display.

4.3.1 Linear Initialization

In this section, is considered the mapping in Section 4.2 to define a linear solution for the viewpoint array parameters associated with a plenoptic camera and the extrinsic parameters for each pose of the calibration grid. The linear solution comprises homography, intrinsic and extrinsic parameters estimation steps.

Homography Estimation. Considering the viewpoint projection matrix $\mathbf{P}^{ij}$ (4.1) with $\mathbf{K}^{ij}$ (4.13) and $\mathbf{t}^{ij}$ (4.18), a point $\mathbf{m} = [x, y, z]^T$ in the object space is projected to a point in the image plane $\mathbf{q} = [k, l]^T$ by

$$\tilde{\mathbf{q}} \sim \mathbf{P}^{ij}\, \tilde{\mathbf{m}} = \mathbf{K}^{ij} \begin{bmatrix} {}^{c}\mathbf{R}_w & {}^{c}\mathbf{t}_w + \mathbf{t}^{ij} \end{bmatrix} \tilde{\mathbf{m}} \tag{4.19}$$

where the symbol $\sim$ denotes equal up to a scale factor. The projection equation (4.19) can be used to estimate the entries of the projection matrix $\mathbf{P}^{ij}$ using a set of 3D points (Appendix C). Alternatively, the coplanar grid points allow to define a world coordinate system such that the z-coordinate is zero. In this context, denoting $\tilde{\mathbf{m}} = [x, y, 1]^T$, one can redefine the projection (4.19) as $\tilde{\mathbf{q}} \sim \mathbf{H}^{ij}\, \tilde{\mathbf{m}}$ where

$$\mathbf{H}^{ij} = \mathbf{K}^{ij} \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & {}^{c}\mathbf{t}_w + \mathbf{t}^{ij} \end{bmatrix} \tag{4.20}$$

is the parametric homography matrix for the viewpoint camera (i, j), and ${}^{c}\mathbf{R}_w = [\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3]$. This matrix can be estimated from the point correspondences $(\tilde{\mathbf{m}}, \tilde{\mathbf{q}})$ using a Direct Linear Transformation (DLT) [1]. Each point correspondence originates two linearly independent equations. The homography matrix has 9 entries to estimate but is defined only up to scale. Thus, $\mathbf{H}^{ij}$ has 8 degrees of freedom needing at least 4 point correspondences to estimate its entries [63]. Assuming a plenoptic camera with N pixels within each microlens and considering an independent estimation of each of the viewpoint cameras' homography matrices, one has $8N$ unknowns to estimate.

A plenoptic camera introduces restrictions on the viewpoint camera array that allows to decrease the number of unknowns to estimate. Namely, the homography matrix $\mathbf{H}^{ij}$ changes among viewpoints as a result of the principal point shift and baseline defined in Section 4.1. Let us consider that $\mathbf{H}^{ij}$ can be defined from the homography matrix $\mathbf{H}^{0}$ associated with the viewpoint coordinates $(i, j) = (0, 0)$ and the homography viewpoint change matrix $\mathbf{A}^{ij}$ by

$$\mathbf{H}^{ij} = \underbrace{\begin{bmatrix} h^0_{11} & h^0_{12} & h^0_{13} \\ h^0_{21} & h^0_{22} & h^0_{23} \\ h^0_{31} & h^0_{32} & h^0_{33} \end{bmatrix}}_{\mathbf{H}^0} + \begin{bmatrix} i & 0 & 0 \\ 0 & j & 0 \\ 0 & 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 0 \end{bmatrix}}_{\mathbf{A}^{ij}} . \tag{4.21}$$

Considering the homography projection of a calibration grid corner $\tilde{\mathbf{m}} = [x, y, 1]^T$ in the object space to the image point $\tilde{\mathbf{q}}$ for the viewpoint camera (i, j), applying the cross product by $\tilde{\mathbf{q}}$ on each side of the projection equation leads to $[\tilde{\mathbf{q}}]_\times \mathbf{H}^{ij}\, \tilde{\mathbf{m}} = \mathbf{0}_{3\times1}$, where $\left[(\cdot)\right]_\times$ is a skew-symmetric matrix that applies the cross product. Using the properties of the Kronecker product [93] and solving for each of the unknown parameters, one obtains

$$\left(\tilde{\mathbf{m}}^T \otimes [\tilde{\mathbf{q}}]_\times\right) \mathbf{T} \begin{bmatrix} \mathbf{h}^0 \\ \mathbf{a}^{ij} \end{bmatrix} = \mathbf{0}_{3\times1} \tag{4.22}$$

where

$$\mathbf{T} = \left[\begin{array}{cc} & \begin{array}{cccccc} i & 0 & 0 & 0 & 0 & 0 \\ 0 & j & 0 & 0 & 0 & 0 \end{array} \\ & \mathbf{0}_{1\times6} \\ & \begin{array}{cccccc} 0 & 0 & i & 0 & 0 & 0 \end{array} \\ \mathbf{I}_{9\times9} & \begin{array}{cccccc} 0 & 0 & 0 & j & 0 & 0 \end{array} \\ & \mathbf{0}_{1\times6} \\ & \begin{array}{cccccc} 0 & 0 & 0 & 0 & i & 0 \\ 0 & 0 & 0 & 0 & 0 & j \end{array} \\ & \mathbf{0}_{1\times6} \end{array}\right] , \tag{4.23}$$

and $\mathbf{h}^0$ and $\mathbf{a}^{ij}$ correspond to vectorizations of the matrix $\mathbf{H}^0$ and $\mathbf{A}^{ij}$

by stacking their columns and removing the zero entries, respectively. The solution $\left[\mathbf{h}^0, \mathbf{a}^{ij}\right]^T$ for the parametric homography matrix can be estimated using Singular Value Decomposition (SVD).

The restrictions introduced by a plenoptic camera allow to represent the parametric homography matrix (4.21) using 15 parameters. According to (4.22), each point correspondence $(\tilde{\mathbf{m}}, \tilde{\mathbf{q}})$ originates three equations with only two being linearly independent. On the other hand, each point in the object space originates N image points, one for each viewpoint camera, assuming that the point is observed in all viewpoint cameras. These pairs provide $2N$ equations that, theoretically, are enough to estimate the parametric homography matrix, assuming that $N \geq 8$. Nonetheless, the restrictions on the viewpoint camera array also originate restrictions on the projections of a point in the object space. Namely, the ray in the image space $\mathbf{\Phi}^{ij} = [i, j, k, l]^T$ associated with an arbitrary viewpoint (i, j) can be described from the ray coordinates $\mathbf{\Phi}^0 = [0, 0, k_0, l_0]^T$ associated with the viewpoint $(i, j) = (0, 0)$ by $\mathbf{\Phi}^{ij} = \mathbf{\Phi}^0 + [i, j, i\beta, j\beta]^T$, where β corresponds to the disparity of the point defined on the VIs. This reduces the number of linearly independent equations originated by a point in the object space to 4. Thus, one needs at least 4 non-collinear points to obtain the entries of the homography matrix $\mathbf{H}^{ij}$.

Denoting $\mathbf{x} = \left[\mathbf{h}^0, \mathbf{a}^{ij}\right]^T$ as the unknown parameters of the homography matrix $\mathbf{H}^{ij}$ and $\mathbf{M}_n = \left(\tilde{\mathbf{m}}^T \otimes [\tilde{\mathbf{q}}]_\times\right) \mathbf{T}$ as the observation matrix associated with the point correspondence $(\tilde{\mathbf{m}}, \tilde{\mathbf{q}})$, the equation (4.22) can be rewritten as $\mathbf{M}_n\, \mathbf{x} = \mathbf{0}_{3\times 1}$. Considering an observation matrix $\mathbf{M}$ obtained from stacking the matrices $\mathbf{M}_n$ of each pair $(\tilde{\mathbf{m}}, \tilde{\mathbf{q}})$, the solution corresponds to a non-zero vector in the null space of $\mathbf{M}$. Since the homography matrix is defined up to a scale factor, one should constraint the solution to $\|\mathbf{x}\|^2 = 1$ leading to the following optimization problem

$$\arg\min_{\mathbf{x}} \|\mathbf{M}\,\mathbf{x}\|^2 \quad \text{s.t.} \quad \|\mathbf{x}\|^2 = 1 \quad . \tag{4.24}$$

In order to obtain an estimate for the homography matrix (4.21), one should consider two practical aspects:

(a) *Data Normalization*: For a DLT it is crucial to normalize the data in order to improve the condition number of the matrix $\mathbf{M}^T\mathbf{M}$ [64]. Thus, one should consider a translation of the image points and the points in the object space so that their centroids are at the origin and the average distances to the origin are equal to $\sqrt{2}$ and $\sqrt{3}$ [63], respectively.

(b) *Computing a Solution in case of a Large Number of Features*: In order to build an over-determined system, having a least squares solution, one should use each projection $\mathbf{q}$ observed in each viewpoint camera for a given point $\mathbf{m}$. Therefore, assuming a plenoptic camera with C viewpoint cameras, a point in the object space generates C point correspondences $(\tilde{\mathbf{m}}, \tilde{\mathbf{q}})$, and consequently $2C$ equations, per raw image. Normally, in a calibration procedure, one uses a calibration grid, with N feature points, that is observed in P different poses. This leads to a tall observation matrix $\mathbf{M}$ with $L = 2C \times N \times P$ rows and 15 columns, *i.e.* one has a high number of observations compared with the number of parameters to estimate. Consequently, using a SVD to obtain the solution to the optimization problem (4.24) is troublesome since this decomposition needs to compute the square matrix $\mathbf{M}\,\mathbf{M}^T$ which requires a prohibitive storage space. Hence, a solution is to perform a QR-Decomposition [51] of the observation matrix $\mathbf{M} = \mathbf{Q}\begin{bmatrix}\mathbf{V} & \mathbf{0}_{(L-15)\times 15}\end{bmatrix}^T$ where $\mathbf{Q}$ is an orthogonal matrix and $\mathbf{V}$ is an upper triangular matrix with size 15×15. This allows to rewrite the optimization problem (4.24) as

$$\arg\min_{\mathbf{x}} \|\mathbf{V}\,\mathbf{x}\|^2 \quad \text{s.t.} \quad \|\mathbf{x}\|^2 = 1 \quad , \tag{4.25}$$

which can be solved using SVD.

Intrinsic and Extrinsic Estimation. The structure of the homography matrix (4.20) in conjunction with the orthogonality and identity of the column vectors of ${}^{c}\mathbf{R}_{w}$ allow to define constraints on the intrinsic parameters as ${\mathbf{h}_1}^T\mathbf{B}^{ij}\,\mathbf{h}_2 = 0$ and ${\mathbf{h}_1}^T\mathbf{B}^{ij}\,\mathbf{h}_1 - {\mathbf{h}_2}^T\mathbf{B}^{ij}\,\mathbf{h}_2 = 0$ [151] where $\mathbf{h}_m$ refers to the m-th column vector of $\mathbf{H}^{ij}$, and the symmetric matrix that describes the image of the absolute conic is defined as $\mathbf{B}^{ij} = {\mathbf{K}^{ij}}^{-T}{\mathbf{K}^{ij}}^{-1}$ [92, 151]. These constraints can be used to obtain the intrinsic parameters independently for each of the viewpoint cameras [151]. Alternatively, one can use the knowledge of the intrinsic matrix $\mathbf{K}^{ij}$ to perform the estimation of a parametric representation of the absolute conic $\mathbf{B}^{ij}$ for a viewpoint camera (i, j) using a minimal number of parameters.

The intrinsic matrix $\mathbf{K}^{ij}$ differs on the principal point for each viewpoint leading to different images of the absolute conic. The principal points change regularly between consecutive viewpoints by the principal point shift $[\Delta u_0, \Delta v_0]^T = \left[-\frac{h_{ui}}{h_{uk}}, -\frac{h_{vj}}{h_{vl}}\right]^T$ which can be used to constraint the parametric representation of $\mathbf{B}^{ij}$. Namely, considering (4.13), $\mathbf{B}^{ij}$ can be defined as

$$\mathbf{B}^{ij} = \mathbf{B}^0 + i\,\mathbf{C}^i + j\,\mathbf{D}^j + i^2\,\mathbf{E}^i + j^2\,\mathbf{F}^j \tag{4.26}$$

with

$$\mathbf{B}^0 = \begin{bmatrix} h_{uk}^2 & 0 & h_u h_{uk} \\ 0 & h_{vl}^2 & h_v h_{vl} \\ h_u h_{uk} & h_v h_{vl} & 1 + h_u^2 + h_v^2 \end{bmatrix}, \tag{4.27}$$

$$\mathbf{C}^i = \begin{bmatrix} 0 & 0 & h_{ui}h_{uk} \\ 0 & 0 & 0 \\ h_{ui}h_{uk} & 0 & 2h_u h_{ui} \end{bmatrix}, \; \mathbf{E}^i = \begin{bmatrix} \multicolumn{3}{c}{\mathbf{0}_{2\times 3}} \\ 0 & 0 & h_{ui}^2 \end{bmatrix}, \tag{4.28}$$

$$\mathbf{D}^j = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & h_{vj}h_{vl} \\ 0 & h_{vj}h_{vl} & 2h_v h_{vj} \end{bmatrix}, \text{ and } \mathbf{F}^j = \begin{bmatrix} \multicolumn{3}{c}{\mathbf{0}_{2\times 3}} \\ 0 & 0 & h_{vj}^2 \end{bmatrix}. \tag{4.29}$$

This allows to define a representation for $\mathbf{B}^{ij}$ using 11 distinct non-zero entries $\mathbf{b}^{ij} = [\, b_{11}, b_{13}, b_{22}, b_{23}, b_{33}, c_{13}, c_{33}, d_{23}, d_{33}, e_{33}, f_{33}\,]^T$ where $(\cdot)_{nm}$ represents the entry in row n and column m of the matrix $(\cdot)$. Considering these parameters, the intrinsic parameters constraints can be redefined as

$$\begin{bmatrix} h_{11}h_{12} & {h_{11}}^2 - {h_{12}}^2 \\ h_{11}h_{32} + h_{12}h_{31} & 2\left(h_{11}h_{31} - h_{12}h_{32}\right) \\ h_{21}h_{22} & {h_{21}}^2 - {h_{22}}^2 \\ h_{21}h_{32} + h_{22}h_{31} & 2\left(h_{21}h_{31} - h_{22}h_{32}\right) \\ h_{31}h_{32} & {h_{31}}^2 - {h_{32}}^2 \\ i\left(h_{11}h_{32} + h_{12}h_{31}\right) & 2i\left(h_{11}h_{31} - h_{12}h_{32}\right) \\ i\left(h_{31}h_{32}\right) & i\left({h_{31}}^2 - {h_{32}}^2\right) \\ j\left(h_{21}h_{32} + h_{22}h_{31}\right) & 2j\left(h_{21}h_{31} - h_{22}h_{32}\right) \\ j\left(h_{31}h_{32}\right) & j\left({h_{31}}^2 - {h_{32}}^2\right) \\ i^2\left(h_{31}h_{32}\right) & i^2\left({h_{31}}^2 - {h_{32}}^2\right) \\ j^2\left(h_{31}h_{32}\right) & j^2\left({h_{31}}^2 - {h_{32}}^2\right) \end{bmatrix}^T \mathbf{b}^{ij} = \mathbf{0}_{2\times 1}\,. \tag{4.30}$$

Normally, each homography generates 2 equations for determining the matrix of the absolute conic image [151]. The parametric homography representation (4.21), representing an arbitrary viewpoint (i, j), generates 6 equations. Nonetheless, only 2 equations are independent regarding the entries of $\mathbf{B}^0$, so one needs to acquire at least 3 calibration grid poses to estimate $\mathbf{b}^{ij}$ defined up to a scale factor.

The intrinsic matrix parameters can be recovered from $\mathbf{B}^{ij}$. More specifically, rewriting the intrinsic matrix $\mathbf{K}^{ij}$ (4.13) as

$$\mathbf{K}^{ij} = \underbrace{\begin{bmatrix} \frac{1}{h_{uk}} & 0 & -\frac{h_u}{h_{uk}} \\ 0 & \frac{1}{h_{vl}} & -\frac{h_v}{h_{vl}} \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{K}^0} + \begin{bmatrix} i & 0 & 0 \\ 0 & j & 0 \\ 0 & 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} \mathbf{0}_{3\times 2} & \begin{matrix} -\frac{h_{ui}}{h_{uk}} \\ -\frac{h_{vj}}{h_{vl}} \\ 0 \end{matrix} \end{bmatrix}}_{\mathbf{G}^{ij}} , \tag{4.31}$$

one can define $\mathbf{B}^0 = \mathbf{K}^{0^{-T}} \mathbf{K}^{0^{-1}}$. This allows to estimate the entries of $\mathbf{K}^0$ using the Cholesky decomposition of $\mathbf{B}^0$ and correcting the scale factor considering $k^0_{33} = 1$. The principal point shift can be estimated considering $\Delta u_0 = -\frac{h_{ui}}{h_{uk}} = -\frac{c_{13}}{b_{11}}$ and $\Delta v_0 = -\frac{h_{vj}}{h_{vl}} = -\frac{d_{23}}{b_{22}}$.

The extrinsic parameters can be estimated once the intrinsic matrix $\mathbf{K}^{ij}$ is known. From (4.20), the rotation matrix ${}^c\mathbf{R}_w = [\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3]$ is recovered considering

$$\mathbf{r}_1 = \lambda \mathbf{K}^{ij^{-1}} \mathbf{h}_1 \ , \ \mathbf{r}_2 = \lambda \mathbf{K}^{ij^{-1}} \mathbf{h}_2 \ , \ \text{and}\ \mathbf{r}_3 = \mathbf{r}_1 \times \mathbf{r}_2 \tag{4.32}$$

with $\lambda = 1/\left\|\mathbf{K}^{ij^{-1}}\mathbf{h}_1\right\| = 1/\left\|\mathbf{K}^{ij^{-1}}\mathbf{h}_2\right\|$. The translation ${}^c\mathbf{t}_w$ and projection center $\mathbf{t}^{ij}$ are recovered solving the following system of equations

$$\lambda \mathbf{h}_3 = \begin{bmatrix} \mathbf{K}^{ij} & -i\mathbf{k}_1 & -j\mathbf{k}_2 \end{bmatrix} \begin{bmatrix} {}^c\mathbf{t}_w \\ h_{si} \\ h_{tj} \end{bmatrix} \tag{4.33}$$

where $\mathbf{k}_m$ corresponds to the m-th column of the parametric intrinsic matrix $\mathbf{K}^{ij}$.

4.3.2 Nonlinear Optimization

In this section, the linear solution is refined and radial distortion [25] is considered on the coordinates (u, v). Namely, the undistorted rays in the object space $\boldsymbol{\Psi}^u = [s, t, u^u, v^u]^T$ are defined from distorted rays in the object space $\boldsymbol{\Psi} = [s, t, u, v]^T$ by

$$\begin{bmatrix} u^u \\ v^u \end{bmatrix} = \left(1 + k_1\, r^2 + k_2\, r^4 + k_3\, r^6\right) \begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} + \begin{bmatrix} b_u \\ b_v \end{bmatrix} \tag{4.34}$$

where $\dot{u} = u^u - b_u$, $\dot{v} = v^u - b_v$, $r^2 = u^2 + v^2$, and $\mathbf{d} = (k_1, k_2, k_3, b_u, b_v)$ defines the distortion vector. In the distortion vector, k_1, k_2 and k_3 are the radial distortion correction coefficients while the vector $[b_u, b_v]^T$ defines the distortion center. In the nonlinear optimization, one minimizes the ray re-reprojection error, *i.e.* the distance between ray and 3D point as defined in [41]. This optimization refines the LFIM parameters $\mathbf{H}$, the extrinsic parameters $\mathbf{R}_p$ (parameterized by Rodrigues formula [48]) and $\mathbf{t}_p$, $p = 1, \dots, P$ where P is the number of poses, and the distortion vector $\mathbf{d}$:

$$\underset{\mathbf{H},\mathbf{R}_p,\mathbf{t}_p,\mathbf{d}}{\arg\min} \sum_{p=1}^{P} \sum_{n=1}^{N_p} \sum_{c=1}^{C} \Lambda\left(\eta_n^c\left(\mathbf{H}, \mathbf{d}\right), \mathbf{R}_p\, \mathbf{m}_n + \mathbf{t}_p\right) \tag{4.35}$$

where N_p corresponds to the number of corners detected on pose p, C corresponds to the number of viewpoint cameras, $\Lambda\left(\cdot\right)$ defines the point-to-ray distance [41], η_n^c defines the undistorted ray coordinates $\mathbf{\Psi}^u$ after mapping the ray in the image space $\mathbf{\Phi}_n^c$ associated with the viewpoint camera c and corner n to the ray in object space (3.1) followed by distortion rectification (4.34). $\mathbf{m}_n$ defines the 3D corner point in the world coordinate system. The nonlinear optimization is solved using the trust-region-reflective algorithm [35], where a sparsity pattern for the Jacobian matrix is provided. The number of parameters over which one optimizes is 8 for the intrinsic parameters, 5 for the lens distortion parameters, and $6P$ for the extrinsic parameters.

4.3.3 Experimental Results

The corner-based calibration methodology proposed is assessed using calibration datasets acquired with commercially available SPCs: the 1^{st}

generation Lytro camera and the most recent Lytro Illum.

Plenoptic cameras acquire images that have higher storage requirements than conventional cameras. Namely, the 1^{st} generation Lytro has a raw image with 3280×3280 pixels that allows to define 9×9 VIs with a resolution of 383×381 pixels after the decoding process described in [41, 44]. On the other hand, the Lytro Illum has a higher spatial and directional resolution in consequence of the higher number of microlenses in the sensor and the higher number of pixels within each microlens. More specifically, the raw image has 7728×5368 pixels that allows to define 15×15 VIs with a resolution of 625×433 pixels after the decoding process described in [41, 44].

1^{st} Generation Lytro State of the Art Comparison. The results of the calibration procedure proposed are compared with the results of the calibrations proposed by Dansereau *et al.* [41] (denoted as *Dans13*) and Bok *et al.* [21] (denoted as *Bok17*). The calibration procedures are applied to publicly available calibration datasets [41] that were obtained using a 1^{st} generation Lytro camera. For this comparison, one considered the Root Mean Square (RMS) of the re-projection error, the ray re-projection error [41], and the reconstruction error, for three stages of the calibration process: the linear solution, the nonlinear refinement, with and without distortion estimation. Three errors are used in this comparison since the re-projection error is the usual error while evaluating pinhole camera calibration procedures but, in plenoptic cameras, the error normally used is the ray re-projection error [21, 41]. In addition, the reconstruction error is used to assess the quality of the reconstruction at the different stages of the calibration. The errors are summarized in Tables 4.2, 4.3, and 4.4. Notice that the values from Bok *et al.* [21] are retrieved directly from their paper.

Comparing the results of the calibration proposed with the results obtained using *Dans13* [41], one can see that the major differences occur at the linear solution. For a subset of 5 poses of these datasets, in the linear

RMS Re-Projection Error (pix)		**Dataset A**	**Dataset B**	**Dataset C**	**Dataset D**	**Dataset E**
Initial	*Dans13* [41]	(10) 1.678 (5) 1.673	(18) 1.687 (5) 1.695	(12) 1.687 (5) 1.671	(10) 1.748 (5) 1.714	(17) 4.290 (5) 4.700
	Bok17* [21]	-	-	-	-	-
	Ours	(10) 0.838 (5) **0.797**	(18) **0.856** (5) 1.035	(12) **0.950** (5) 0.953	(10) 0.965 (5) **0.790**	(17) 0.840 (5) **0.627**
Optimized	*Dans13* [41]	(10) 0.435 (5) 0.372	(18) 0.406 (5) 0.429	(12) 0.402 (5) **0.392**	(10) 0.404 (5) 0.461	(17) 0.218 (5) 0.185
	Bok17* [21]	-	-	-	-	-
	Ours	(10) 0.427 (5) **0.366**	(18) **0.405** (5) 0.435	(12) 0.420 (5) **0.392**	(10) **0.389** (5) 0.489	(17) 0.219 (5) **0.177**
Optimized (with Distortion)	*Dans13* [41]	(10) 0.226 (5) 0.221	(18) 0.191 (5) 0.240	(12) 0.161 (5) 0.164	(10) 0.150 (5) 0.163	(17) 0.190 (5) 0.153
	Bok17* [21]	(5) 0.374	(9) 0.259	-	-	(14) 0.274
	Ours	(10) 0.226 (5) **0.211**	(18) **0.179** (5) 0.194	(12) **0.156** (5) 0.159	(10) **0.145** (5) 0.163	(17) 0.134 (5) **0.127**

Table 4.2: RMS re-projection error in pixels for three stages of the calibration procedure: linear solution, and nonlinear refinement with and without distortion estimation. The number of poses P considered for the calibration is denoted as (P). The symbol * indicates that the values reported are retrieved directly from the corresponding paper.

RMS Ray Re-Projection Error (mm)		**Dataset A**	**Dataset B**	**Dataset C**	**Dataset D**	**Dataset E**
Initial	*Dans13** [41]	(10) 3.200	(18) 5.060	(12) 8.630	(10) 5.920	(17) 13.800
	Dans13 [41]	(10) 0.577 (5) 0.627	(18) 0.603 (5) 0.570	(12) 1.036 (5) 0.974	(10) 1.231 (5) 1.081	(17) 8.900 (5) 11.970
	Bok17* [21]	-	-	-	-	-
	Ours	(10) **0.307** (5) 0.314	(18) **0.341** (5) 0.353	(12) 0.609 (5) **0.593**	(10) 0.640 (5) **0.478**	(17) **1.657** (5) 1.709
Optimized	*Dans13** [41]	(10) 0.146	(18) 0.148	(12) 0.255	(10) **0.247**	(17) **0.471**
	Dans13 [41]	(10) 0.154 (5) 0.145	(18) 0.147 (5) **0.139**	(12) 0.260 (5) **0.245**	(10) 0.260 (5) 0.268	(17) 0.485 (5) 0.546
	Bok17* [21]	-	-	-	-	-
	Ours	(10) 0.151 (5) **0.143**	(18) 0.143 (5) **0.139**	(12) 0.271 (5) 0.247	(10) 0.251 (5) 0.277	(17) 0.489 (5) 0.532
Optimized (with Distortion)	*Dans13** [41]	(10) **0.084**	(18) **0.063**	(12) 0.106	(10) **0.105**	(17) **0.363**
	Dans13 [41]	(10) 0.085 (5) 0.086	(18) 0.066 (5) 0.069	(12) 0.104 (5) **0.102**	(10) 0.116 (5) 0.117	(17) 0.390 (5) 0.456
	Bok17* [21]	(5) 0.108	(9) 0.071 (5) 0.072	-	-	(14) 0.492 (5) 0.454
	Ours	(10) 0.085 (5) 0.085	(18) 0.066 (5) 0.066	(12) 0.103 (5) 0.103	(10) 0.114 (5) 0.116	(17) 0.393 (5) 0.457

Table 4.3: RMS ray re-projection error in mm for three stages of the calibration procedure: linear solution, and nonlinear refinement with and without distortion estimation. As in Table 4.2, (P) denotes P poses, and * indicates values retrieved from related work.

RMS Reconstruction Error (mm)		**Dataset A**	**Dataset B**	**Dataset C**	**Dataset D**	**Dataset E**
Initial	*Dans13* [41]	(10) 2100.536 (5) 3139.904	(18) 325.215 (5) 203.736	(12) 1293.985 (5) 1397.874	(10) 844.038 (5) 517.783	(17) 2702.292 (5) 3370.725
	Ours	(10) **3.039** (5) 3.904	(18) **6.212** (5) 8.023	(12) 14.899 (5) **12.558**	(10) **20.751** (5) 25.316	(17) **79.681** (5) 102.281
Optimized	*Dans13* [41]	(10) **3.370** (5) 3.627	(18) 4.367 (5) **3.112**	(12) 10.174 (5) 10.607	(10) 15.050 (5) 12.401	(17) **123.728** (5) 253.959
	Ours	(10) 3.747 (5) 3.682	(18) 4.516 (5) 3.927	(12) 10.229 (5) **8.277**	(10) 15.168 (5) **12.216**	(17) 142.231 (5) 187.750
Optimized (with Distortion)	*Dans13* [41]	(10) 4.408 (5) 4.283	(18) 4.652 (5) **4.415**	(12) 9.995 (5) 8.007	(10) 15.425 (5) 13.051	(17) **135.851** (5) 179.697
	Ours	(10) 4.443 (5) **4.256**	(18) 4.706 (5) **4.415**	(12) 9.976 (5) **7.932**	(10) 15.553 (5) **12.700**	(17) 138.968 (5) 183.037

Table 4.4: RMS reconstruction error in mm for three stages of the calibration procedure: linear solution, and nonlinear refinement with and without distortion estimation. As in Tables 4.2 and 4.3, (P) denotes P poses.

solution stage, one can see that the re-projection error of *Dans13* [41] is at least 1.63 times higher, the ray re-projection error is at least 1.61 times higher, and the reconstruction error is at least 20.45 times higher. These differences between the two calibration methods are even greater considering the complete datasets. This confirms that the proposed method for the linear solution outperforms the state of the art.

Comparing with Bok *et al.* [21], the proposed calibration obtains smaller re-projection and ray re-projection errors using the complete datasets. Namely, the re-projection error is 1.44 smaller, and the ray re-projection error is 1.25 times smaller. Only Dataset B presents a similar performance to the calibration proposed. Considering a subset of 5 poses, the ray re-projection errors obtained for Bok *et al.* [21] are similar with the ones of the calibration proposed with the exception of Dataset A that exhibits an error 1.26 times higher.

The results obtained show that the reduced LFIM (4.17) does not degrade the performance of the calibration procedure. In this representation, the position (s, t) of the ray can be represented using only the viewpoint coordinates (i, j) which allows to represent the rays with a minimal number of sub-camera apertures.

Calibration Precision with Number of Poses. As in the calibration of conventional pinhole cameras, the redundancy and accuracy of calibration data is a key factor for attenuating the effect of calibration data noise into the calibration precision. Dansereau *et al.* [41] considered the influence of different sizes of calibration patterns while Bok *et al.* [21] considered the influence of different number of poses for the 1^{st} generation Lytro camera.

The Lytro Illum camera is more recent than the 1^{st} generation Lytro camera, and its specifications indicate improvements in almost all technical aspects. Thus, one wants to assess the influence of different sizes of the calibration patterns and different number of poses for a Lytro Illum camera. For this purpose, one acquired new calibration datasets with a Lytro Illum camera using two calibration grids with different sizes: 8×6 grid of 211×159 mm with approximately 26.5 mm cells (denoted as Illum-1), and 20×20 grid of 121.5×122 mm with approximately 6.1 mm cells (denoted as Illum-2). Each dataset acquired is composed of 66 fully observable poses of the calibration pattern. Care was taken to avoid changing the focal settings of the camera.

The higher number of poses acquired allow to define several subsets for calibration which allow a statistical analysis of the results. Therefore, in order to evaluate the precision of the calibration, one repeated 20 times the calibration procedure. Each calibration involves $k = 2, \ldots, 20$ pattern poses, randomly selected from the full calibration dataset. The calibration procedure proposed is compared with the methodology [41] (denoted as *Dans13*).

In Figure 4.5, the mean and Standard Deviation (STD) obtained for the re-projection error, the ray re-projection error, and the reconstruction error with the number of poses for Dataset Illum-1 and Illum-2 are depicted. This figure shows that the errors are similar for both calibration methods after nonlinear refinement. However, for the linear solution, the calibration proposed obtains smaller errors using 3 or more calibration pattern poses. Additionally, to evaluate the precision associated with the estimate of each LFIM parameter with the number of poses, one shows in Figure 4.6 the mean and STD obtained for each parameter of LFIM for Datasets Illum-1 and Illum-2. Notice that the calibration *Dans13* [41] obtains a LFIM with 12 non-zero entries while the method proposed obtains LFIM with 8 non-zero entries. For comparing the parameters, one transformed the LFIM obtained by *Dans13* as defined in Section 4.2. According to the results, one needs 9 poses using *Dans13* [41] and 8 poses using the proposed calibration for Dataset Illum-1 for having a deviation smaller than 3% of the mean value. In Dataset Illum-2, one needs 9 poses using *Dans13* [41] and the proposed calibration.

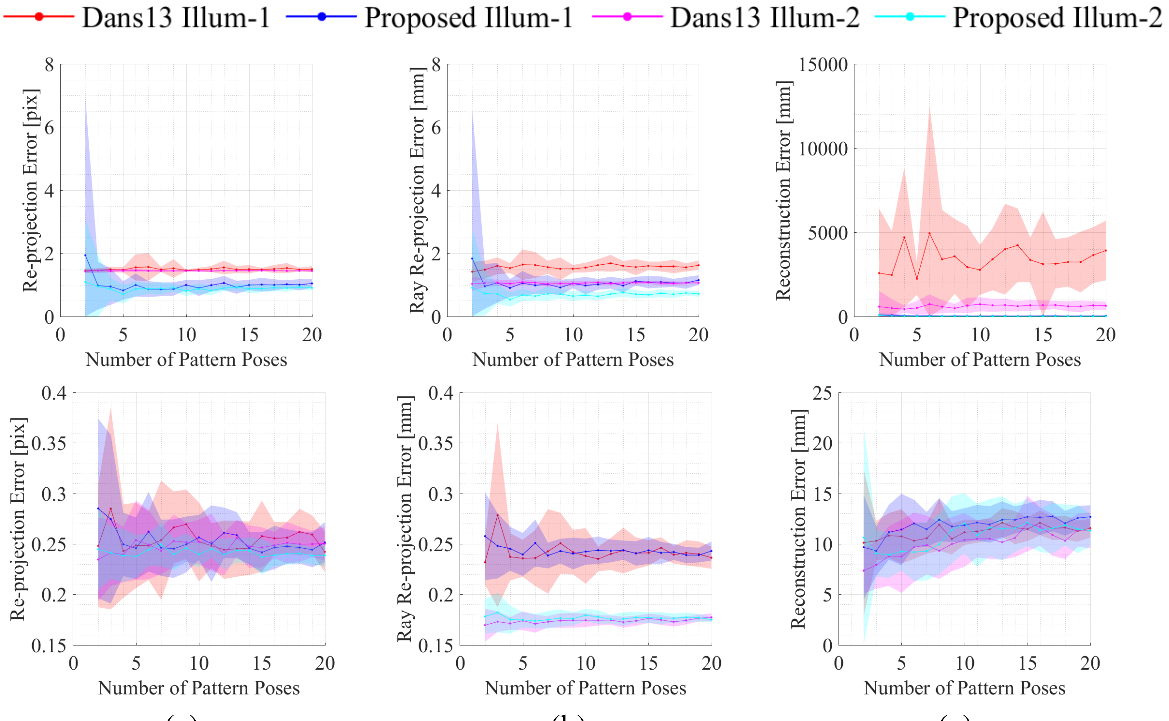
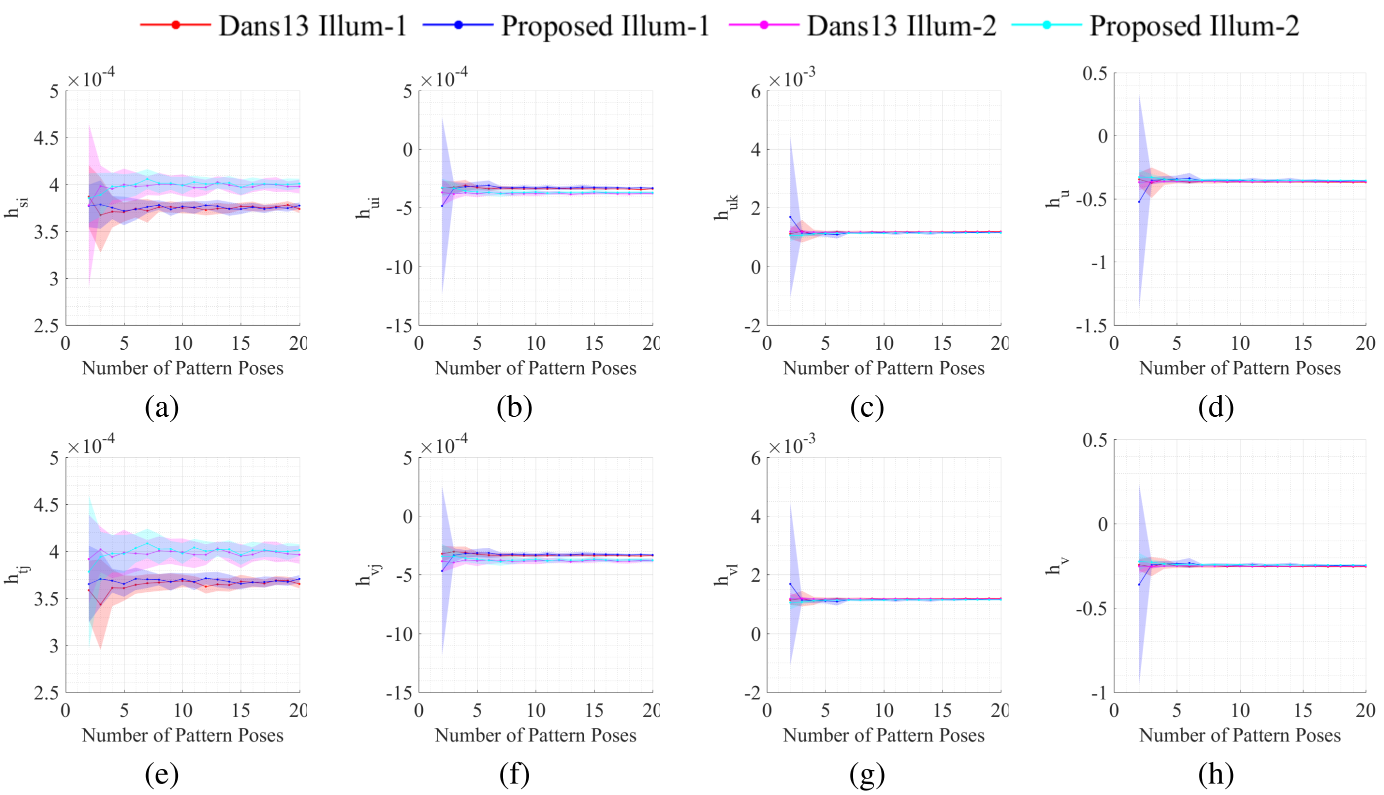


Figure 4.5: RMS errors obtained using the calibration proposed (in blue and cyan for Dataset Illum-1 and Illum-2, respectively) and the calibration *Dans13* [41] (in red and magenta for Dataset Illum-1 and Illum-2, respectively): re-projection error (**a**), ray re-projection error (**b**), and reconstruction error (**c**). The first row depicts the errors obtained for the linear solution and the second row depicts the errors obtained for the nonlinear refinement with distortion estimation.

Let us also consider the statistical analysis of the difference between the estimates at the initial and final stages of the calibration process for each of the entries of the LFIM. The mean and STD values for Dataset Illum-1 and Illum-2 are depicted in Figure 4.7. This figure shows that



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

(e) (f) (g) (h)

Figure 4.6: Precision of the LFIM parameters after nonlinear refinement with distortion estimation using the calibration proposed (in blue and cyan for dataset Illum-1 and Illum-2, respectively) and the calibration *Dans13* [41] (in red and magenta for dataset Illum-1 and Illum-2, respectively): h_{si} **(a)**, h_{ui} **(b)**, h_{uk} **(c)**, h_u **(d)**, h_{tj} **(e)**, h_{vj} **(f)**, h_{vl} **(g)**, and h_v **(h)**. The mean values are represented by the solid lines and the STD by the shaded areas.

the calibration proposed allows to obtain an initial solution that is closer to the solution at the final stage of the calibration procedure. Namely, the proposed calibration allows to estimate more precisely the entries related with the baseline and the principal point shift (Figure 4.7.a-b and 4.7.e-f). For the remaining entries, the performance is similar for both calibration methods.

The calibration proposed is applied to a set of 10 randomly sampled poses and on a reduced set of 5 poses to evaluate the quality of the calibration. For comparison purposes, these sets of poses are also calibrated using the calibration described by Dansereau *et al.* [41]. For this comparison, one considered the RMS of the re-projection error, the ray reprojection error [41], and the reconstruction error, for three stages of the calibration process: the linear solution, the nonlinear refinement, with

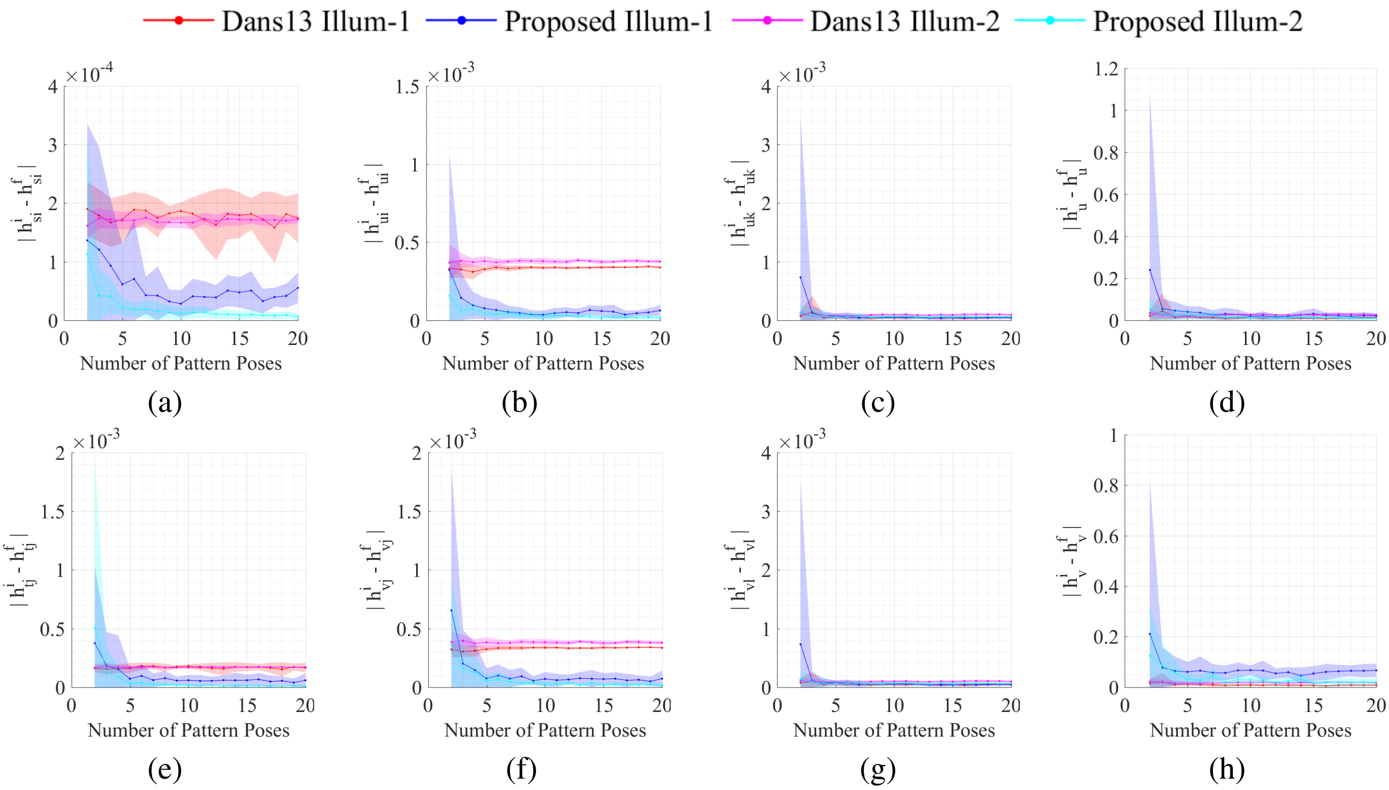


Figure 4.7: Difference between the estimated LFIM parameters at the initial and final stages of the calibration proposed (in blue and cyan for dataset Illum-1 and Illum-2, respectively) and the calibration *Dans13* [41] (in red and magenta for dataset Illum-1 and Illum-2, respectively): h_{si} **(a)**, h_{ui} **(b)**, h_{uk} **(c)**, h_u **(d)**, h_{tj} **(e)**, h_{vj} **(f)**, h_{vl} **(g)**, and h_v **(h)**. The mean values are represented by the solid lines and the STD by the shaded areas.

and without distortion estimation. The errors are summarized in Table 4.5.

The re-projection, ray re-projection and reconstruction errors are similar after the refinement of the linear solution for both calibration methods. Also, the lower number of poses does not appear to change the errors significantly after the nonlinear optimization. The accuracy of the calibration proposed can be seen from the maximum errors obtained at the final stage of the calibration: the re-projection error has sub-pixel accuracy (below 0.29 pixels), the ray re-projection error is below 0.26 mm, and the reconstruction error is below 12 mm.

For the linear solution, the re-projection and ray re-projection errors are similar for the Dataset Illum-1. However, for the Dataset Illum-2, one can see that these errors are smaller for the calibration proposed.

RMS Re-Projection Error (pix)		**Illum-1 10 poses**	**Illum-1 5 poses**	**Illum-2 10 poses**	**Illum-2 5 poses**
Initial	Dans13 [41]	1.463	1.501	1.480	1.485
	Ours	1.659	1.400	**0.922**	1.249
Optimized	Dans13 [41]	**0.320**	0.428	0.418	0.429
	Ours	0.332	0.429	0.421	0.446
Optimized (with Distortion)	Dans13 [41]	**0.235**	0.288	0.293	0.288
	Ours	0.249	0.284	0.263	0.270

RMS Ray Re-Projection Error (mm)		**Illum-1 10 poses**	**Illum-1 5 poses**	**Illum-2 10 poses**	**Illum-2 5 poses**
Initial	Dans13 [41]	1.698	1.516	0.965	0.891
	Ours	1.813	1.623	**0.617**	0.776
Optimized	Dans13 [41]	0.322	0.342	**0.245**	0.255
	Ours	0.334	0.347	0.247	0.261
Optimized (with Distortion)	Dans13 [41]	0.241	0.239	0.168	0.173
	Ours	0.254	0.243	**0.166**	0.172

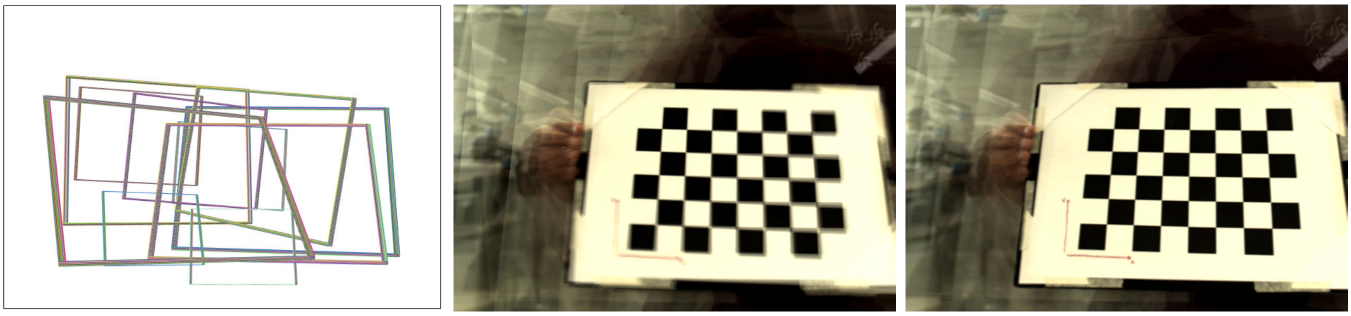
RMS Reconstruction Error (mm)		**Illum-1 10 poses**	**Illum-1 5 poses**	**Illum-2 10 poses**	**Illum-2 5 poses**
Initial	Dans13 [41]	3483.898	1553.119	304.433	199.785
	Ours	37.594	18.717	**7.560**	9.626
Optimized	Dans13 [41]	13.625	9.046	8.126	**6.496**
	Ours	13.747	10.563	8.242	6.914
Optimized (with Distortion)	Dans13 [41]	10.680	10.255	7.070	**5.968**
	Ours	11.939	10.526	6.850	6.250

Table 4.5: RMS errors for three stages of the calibration procedure: linear solution, and nonlinear refinement with and without distortion estimation.

Additionally, the reconstruction error is considerably higher for the calibration proposed by Dansereau *et al.* [41] regardless of the dataset considered. More specifically, the reconstruction error is at least 20 times higher than the one obtained using the calibration proposed.

The major difference between *Dans13* [41] and the proposed method corresponds to the linear solution. The linear solution used by Dansereau *et al.* [41] does not consider any constraint to obtain the homographies between each viewpoint and the calibration grid pose, *i.e.* for a Lytro Illum camera one computes $P \times 15 \times 15$ homographies where P corresponds to the number of calibration grid poses. On the other hand, the proposed method exploits the geometry of the viewpoint camera array to estimate a parametric homography matrix that characterizes the SPC for each calibration grid pose, *i.e.* P homographies are computed. Additionally, in Dansereau *et al.* [41], the principal point shift is assumed to be zero on the linear solution and is only estimated during the nonlinear refinement. The more accurate representation of the viewpoint camera array by the calibration proposed allows to obtain an initial solution that is closer to the final one.

Finally, let us evaluate the quality of the estimated poses and of the distortion model. For the estimated poses, one considered an image that corresponds to the mean of the intensity values after warping all VIs using the homography matrix estimated from LFIM entries for all calibration grid poses. The images for Dataset Illum-1 for the initial and final stages of the calibration process are depicted in Figure 4.8. Notice that in the final stage of the calibration, the edges of the calibration grid are not blurred.



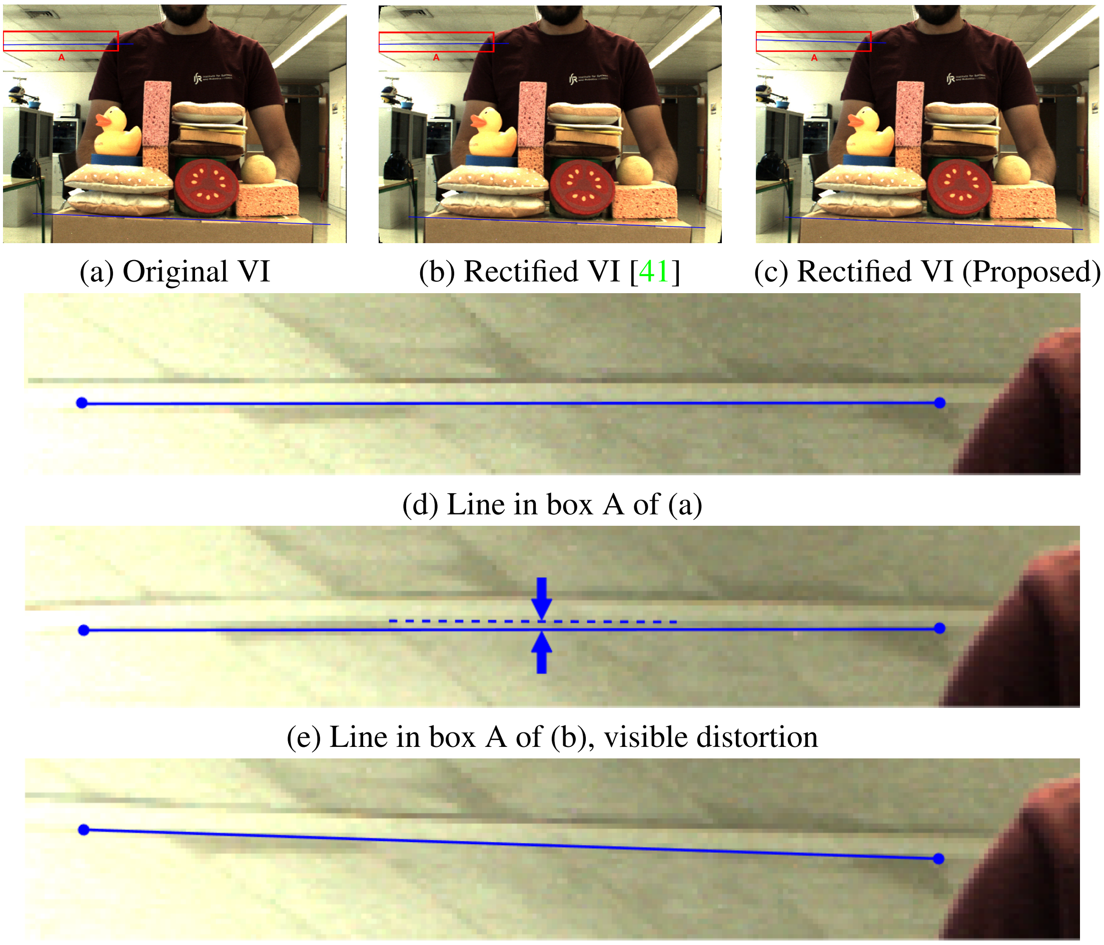
(a) All imaged chessboard poses

(b) Merged after linear solution

(c) Merged after nonlinear solution

Figure 4.8: Mean intensity values for all VIs warped using the homography matrix estimated from LFIM entries for all 10 calibration grid poses for Dataset Illum-1. **(a)** depicts the calibration pattern limits for the different calibration grid poses without homography correction. **(b)** depicts the images obtained for the linear solution and **(c)** depicts the images obtained for the nonlinear refinement with distortion estimation.

For the distortion model, one has rectified the LF of a scene that was not considered for the calibration using the distortion parameters estimated with the calibration proposed and *Dans13* [41]. The central VI of the rectified LF considering the results of the calibration on a subset of 10 poses for Dataset Illum-2 is presented in Figure 4.9. The two approaches behave similarly in rectifying the straight lines in the foreground of the scene (Figure 4.9.b-c). Notice that for *Dans13* [41] (Figure 4.9.e), the straight lines in the background are distorted in the rectification. Nonetheless, the rectification using the parameters estimated with the calibration proposed allows to maintain straight lines in the background and in the foreground (Figure 4.9.f).



(a) Original VI (b) Rectified VI [41] (c) Rectified VI (Proposed)

(d) Line in box A of (a)

(e) Line in box A of (b), visible distortion

(f) Line in box A of (c), lesser visible distortion

Figure 4.9: Distortion rectification using the distortion parameters estimated with *Dans13* [41] ((**b**) and (**e**)) and the calibration proposed ((**c**) and (**f**)) for the Dataset Illum-2. (**a**) depicts the original central VI while (**d**)-(**f**) correspond to zooms of the red boxes A. Blue rulings were added to aid in the visual confirmation of the straight lines after rectification.

Viewpoint Camera Array. The characterization of the viewpoint camera array for a 1^{st} generation Lytro camera (Dataset B [41]) and for a Lytro Illum camera are presented in Table 4.6. The viewpoint array parameters are obtained from the LFIM estimated at the final stage of the calibration procedure. The camera array of the 1^{st} generation Lytro camera is characterized by a unitary baseline length $\left\|\mathbf{t}^{ij}\right\| = \sqrt{\Delta x_0^2 + \Delta y_0^2}$ of 0.37 mm. Considering the 9×9 viewpoint cameras, the maximum baseline length that can be defined is 2.97 mm. The non-zero principal point shift shows that the principal point is different for each viewpoint camera. This gives a zero disparity 3D plane, *i.e.* the plane in focus Ω, positioned approximately at 0.29 m for Dataset B [41].

Table 4.6 shows that the viewpoint camera array for Lytro Illum has a scale factor and baseline greater than the 1^{st} generation Lytro. The estimated unitary baseline length for the Lytro Illum is 0.52 mm and the maximum baseline length considering the 15×15 viewpoint cameras is 7.33 mm. Thus, the unitary baseline for the Lytro Illum is 1.41 times higher than the 1^{st} generation counterpart. If one considers the full camera array, the maximum baseline for the Lytro Illum is 2.46 times higher. The increased scale factor is justified by the higher spatial resolution of the raw image (assuming the sensor size remains constant). Notice also the non-zero estimate for the principal point shift that defines a plane in focus Ω positioned approximately at 1.09 m. This estimate confirms that the principal point is different for each viewpoint camera and consequently the epipolar geometry for the SPC corresponds to the one defined in Section 4.1.3.

$\mathbf{P}^{ij}$	k_u	k_v	u_0	v_0	Δx_0	Δy_0	Δu_0	Δv_0
Dataset B [41]	545.84	547.10	188.94	189.03	-0.00027	-0.00026	0.51	0.49
Illum-1 10 poses	841.55	840.40	310.76	214.68	-0.00036	-0.00038	0.28	0.29

Table 4.6: Parameters to describe the viewpoint camera arrays of commercially available SPCs: Lytro Illum and 1^{st} generation Lytro cameras.

Notice that the viewpoint cameras are virtual cameras so the properties associated with the camera arrays like baseline, scale factor and principal point shift will vary with different zoom and focus settings of the SPC (Section 4.4.2).

The viewpoint camera array for a Lytro Illum camera is depicted in Figure 4.1.d. Figure 4.1.b-d shows the raw image, reconstructed structure and the viewpoint camera array that characterizes a Lytro Illum camera. The reconstruction, Figures 4.1.c and 4.1.d, is obtained from disparities estimated with the structure tensor [95], which are converted to depth values, in metric units, based on the calibration parameters. The calibration parameters were extracted from the LFIM obtained at the final stage of the proposed calibration procedure using a subset of 10 poses of Illum-1.

4.4 Metadata-based Calibration

The camera models proposed for SPCs [21, 41, 150] are approximations of the real setup by considering the main lens as a thin lens and the microlenses as pinholes. There can be more complex models to describe the real setup. The SPC manufacturer provides metadata regarding the camera optical settings that help describing the camera. Namely, the metadata provided include the main lens focal length which is considered in [21, 41] to model the refraction of the rays by the main lens. On the other hand, the metadata also includes the distance at which a point is always in focus by the microlenses. Nonetheless, the assumption of pinhole like microlenses do not allow to incorporate directly this additional information on the camera models [21, 41, 150].

The calibration procedures for SPCs in the literature [21, 41, 150] and the one proposed in Section 4.3 do not consider the information provided by the camera manufacturer as metadata and therefore rely completely on the acquisition of a dataset with a calibration pattern to estimate the camera model parameters for a specific zoom and focus settings. Thus, in this section, one identifies the relationships among the optical parameters provided as metadata as well as the relationships between these optical parameters and the entries of the LFIM $\mathbf{H}$ (4.17) for different zoom and focus settings of the camera. The relationships obtained are used to estimate the LFIM parameters based on the metadata parameters for a specific zoom and focus setting without having to acquire a novel calibration dataset (Figure 4.10).

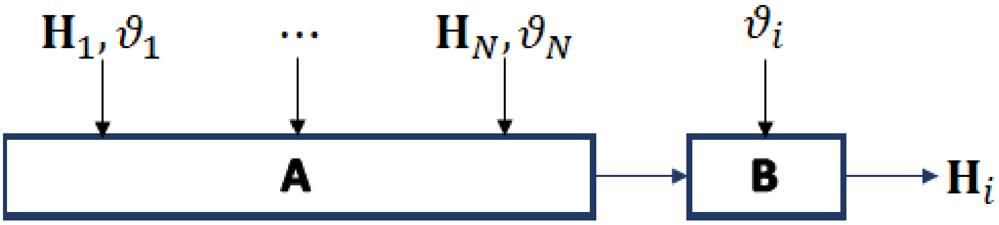


Figure 4.10: Representation of an SPC based on meta-parameters provided in the images metadata. In step A, the affine functions $\mathrm{a}(f)$, $\mathrm{b}(f)$, $\mathrm{c}(f)$, $\mathrm{d}(f)$, $\mathrm{e}(\lambda_\infty)$, and $\mathrm{g}(\lambda_\infty)$ are estimated using several calibration datasets with different zoom and focus settings. These datasets are used to relate the entries of the LFIM $\mathbf{H}_{(\cdot)}$ and the meta-parameters $\vartheta_{(\cdot)}$ (Section 4.4.2). In step B, the LFIM $\mathbf{H}_i$ is estimated for an arbitrary zoom and focus settings using only the meta-parameters ϑ_i of a given image and without acquiring a calibration dataset for that specific zoom and focus settings.

4.4.1 Camera Metadata Parameters

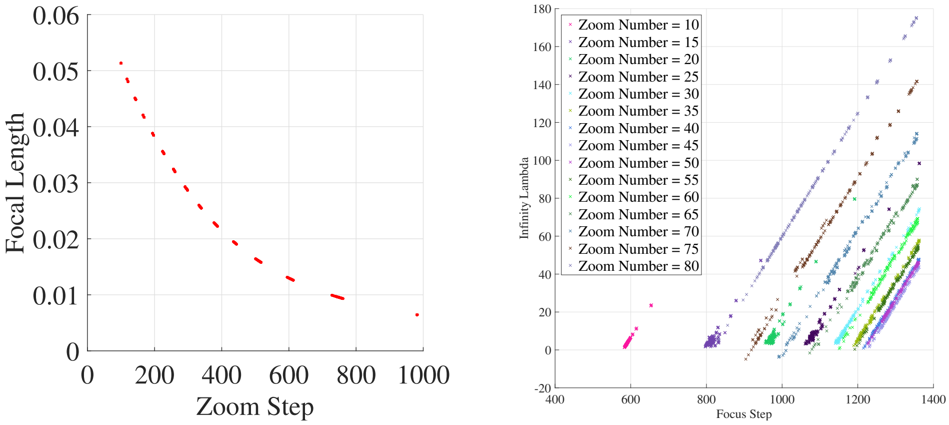
The metadata parameters (meta-parameters), provided by the camera manufacturer with the images acquired, are retrieved from the camera hardware. Here, one focus on the information that refers to the image sensor, main lens and microlens array. More specifically, meta-parameters that change with the zoom and focus settings of the camera, *i.e.* the main lens world focal plane [107].

In order to identify and analyze the camera meta-parameters depending on zoom and focus settings, one compared the camera meta-parameters with the depth of a target object in the world coordinate system and computed the Pearson correlation coefficient among the different meta-parameters [49]. For this experiment, one acquired a set of images by placing the target object parallel to the encasing of the camera and at a regular spacing of 0.05 m from the camera. The target object depths ranged from 0.05 m to 2.00 m. The zoom number (number that appears on the interface of the camera) was changed also at a regular interval of 0.5 between 1.0 and 8.0. At each of these configurations, *i.e.* for a fixed target object depth and fixed zoom number, a total of 5 images were taken autofocusing on the target object.

In this experimental analysis, one identifies five parameters that vary with the main lens world focal plane: zoom step (zoom-stepper motor position), focus step (focus-stepper motor position), main lens focal length, infinity lambda, and f-number. The infinity lambda can be associated with the focus settings of the microlenses. Namely, the infinity lambda corresponds to the distance in front of the microlens array that is in focus at infinity. However, the microlenses optical settings are fixed. The optical settings are changed by modifying the main lens or the complex of lenses that compose the main lens. Thus, the infinity lambda describes the combined optical setup of the microlenses and main lens. On the other hand, the f-number is not used in the definition of the intrinsic parameters of a camera and it is normally described as the ratio f/D where f is the focal length and D is the diameter of the entrance pupil.

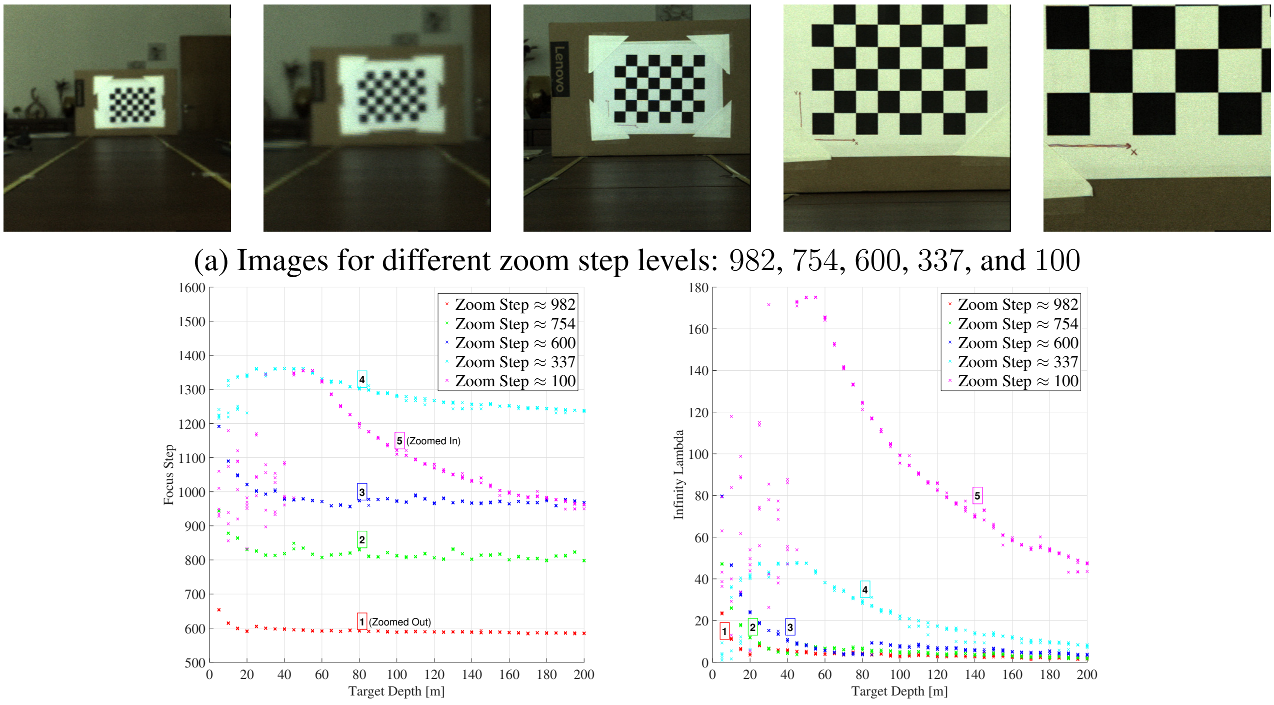
The parameters zoom and focus step represent, up-to an affine transformation, optical parameters information. Namely, the zoom step is related with the focal length of the main lens (Figure 4.11.a) (correlation of 93.16%), and the focus step for a fixed zoom is related with the infinity lambda parameter (Figure 4.11.c) (correlation of 99.54%). In fact, representing the focal length, infinity lambda and target object depth (Figure 4.12.c), one finds a similar behavior to the one depicted in Figure 4.12.b that represents the zoom step, focus step and target object depth. This reduces the relevant metadata parameters to two, the focal length and the infinity lambda.

Figure 4.11 shows that for a particular focal length (zoom step) configuration, there is a depth at which the camera is not able to autofocus on the target object (the infinity lambda and focus step do not change) and, consequently, the world focal plane does not change. This failure in focusing the target object occurs due to poor detail of the features in the image. The camera is only capable of focusing the target object, *i.e.* changing the world focal plane, if the focal length is decreased (zoom step is increased). Additionally, for extreme conditions of the operating range of the plenoptic camera, for example considering zoom step close



(a) (b)

Figure 4.11: Relationships among camera parameters provided on images metadata. The meta-parameters were obtained experimentally by fixing the zoom number and autofocusing the camera to a target object placed at different depths. The zoom step **(a)** is related with the focal length of the main lens. The focus step **(b)** is related with the infinity lambda parameter. The zoom number corresponds to the number that appears on the interface of the camera.



(a) Images for different zoom step levels: 982, 754, 600, 337, and 100

(b) Focus step vs. Target depth (c) Infinity lambda vs. Target depth

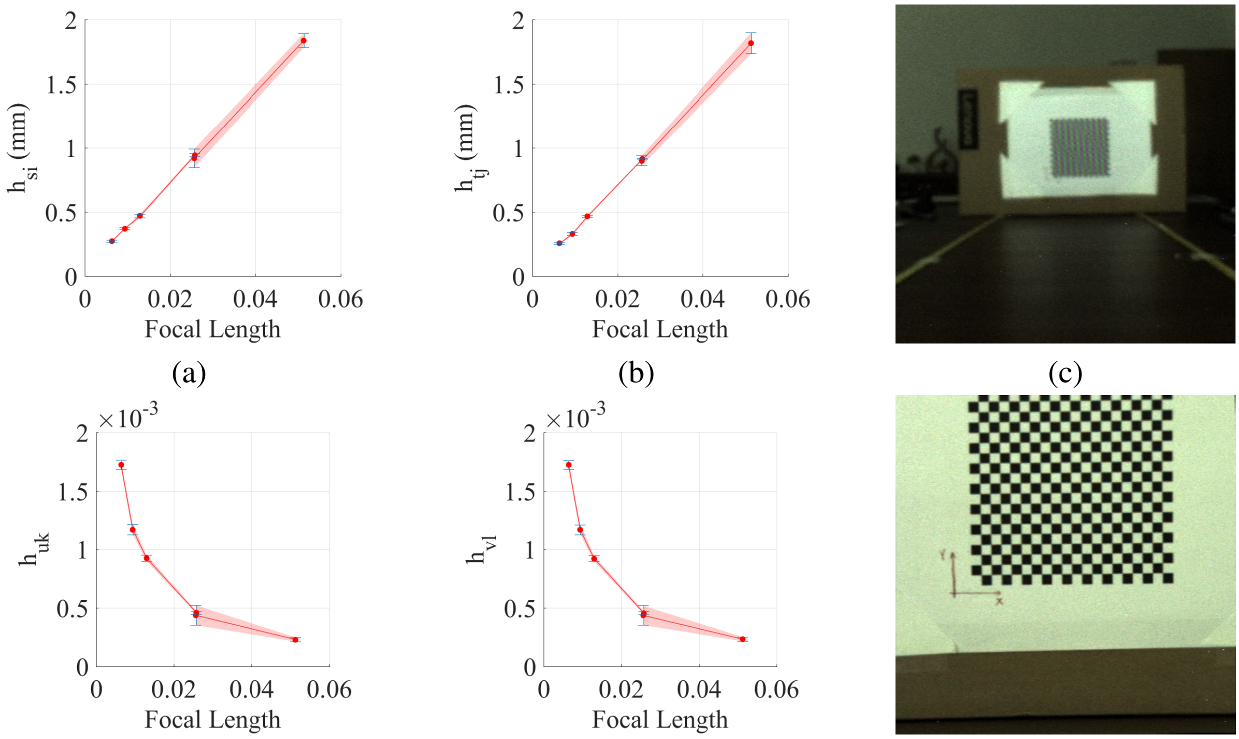
Figure 4.12: Meta-parameters vs. Target depth. **(a)** represents the target object at depth 1.5 m for the different zoom steps. **(b)** represents the focus step with the depth of a target object for a selection of zoom steps. **(c)** represents the infinity lambda with the depth of a target object for a selection of zoom steps (or equivalently, focal lengths).

to 100 and target object depths smaller than 0.4 m, one can see that the infinity lambda changes arbitrarily among the several attempts to autofocus on the target object depth. This results in images with no sharp objects. This situation also corresponds to a failure on focusing the target object. For focusing at these target object depths, the focal length should be increased (zoom step should be decreased). This shows that the world focal plane is defined by a combination of the focal length and the infinity lambda parameters.

4.4.2 Metadata Parameters vs. LFIM

The LFIM depends on the optical settings of the camera. The derivation in Section 3.5 indicates that the LFIM parameters change with the main lens focal length included in the images metadata. However, the assumption of microlenses as pinholes do not allow to introduce the concept of focus at infinity as a parameter of the LFIM. Thus, let us provide the relationships between the LFIM parameters and the camera parameters provided on the images metadata.

In order to evaluate these relationships, one needs multiple calibration datasets acquired under different zoom and focus settings (Section 4.5.1). These datasets were acquired using a 1^{st} generation Lytro camera and are summarized in Table 4.10. For establishing the relationships, were considered 10 poses randomly selected from the acquired calibration pattern poses to estimate the LFIM $\mathbf{H}$ (4.17) and repeated this procedure 15 times to get the mean and STD values. Representing the entries of the LFIM and computing their Pearson correlation coefficients [49] against the focal length and infinity lambda, was found that the entries h_{si} and h_{tj}, which are related to the viewpoint camera baseline, exhibit an affine relationship with the focal length (Figure 4.13.a-b) with a correlation coefficient of 99.97% and 99.98%, respectively. The entries h_{uk} and h_{vl}, which are related with the scale factors, exhibit a nonlinear relationship with the focal length (Figure 4.13.d-e) with a correlation coefficient of 84.94% and 84.75%, respectively. The remaining entries do not exhibit a correlation with any of the metadata parameters provided.



(a) (b) (c)

(d) (e) (f)

Figure 4.13: Relationships of the LFIM entries with the focal length. The entries related with the viewpoint camera baseline **(a)-(b)**, and with the scale factors **(d)-(e)** are represented against the focal length. The target object is depicted at 1 m with different focal lengths (0.0064 **(c)** and 0.0256 **(f)**).

Alternatively, let us analyze the viewpoint array parameters against the camera metadata parameters. Considering the entries of the intrinsic matrix $\mathbf{K}^{ij}$ (4.13), one founds that the scale factors $k_u = 1/h_{uk}$ and $k_v = 1/h_{vl}$ exhibit an affine relationship with the focal length (Figure 4.14.a-b) with a correlation coefficient of 99.82% and 99.81%, respectively. The principal point shifts h_{ui}/h_{uk} and h_{vj}/h_{vl} have an affine relationship with the infinity lambda (Figure 4.14.c-d) with a correlation coefficient of 99.55% and 99.83%, respectively. The principal point $\left[h_u/h_{uk}, h_v/h_{vl}\right]^T$ continues not having any relationship with the metadata parameters. Hence, the transformation of the LFIM to a pinhole like representation allows to simplify the relationships with the parameters provided by the manufacturer on the metadata of the images acquired.

In summary, denoting $[c_u, c_v]^T$ as the principal point and $\vartheta = (f, \lambda_\infty)$ where f is the main lens focal length and λ_∞ is the infinity lambda, one

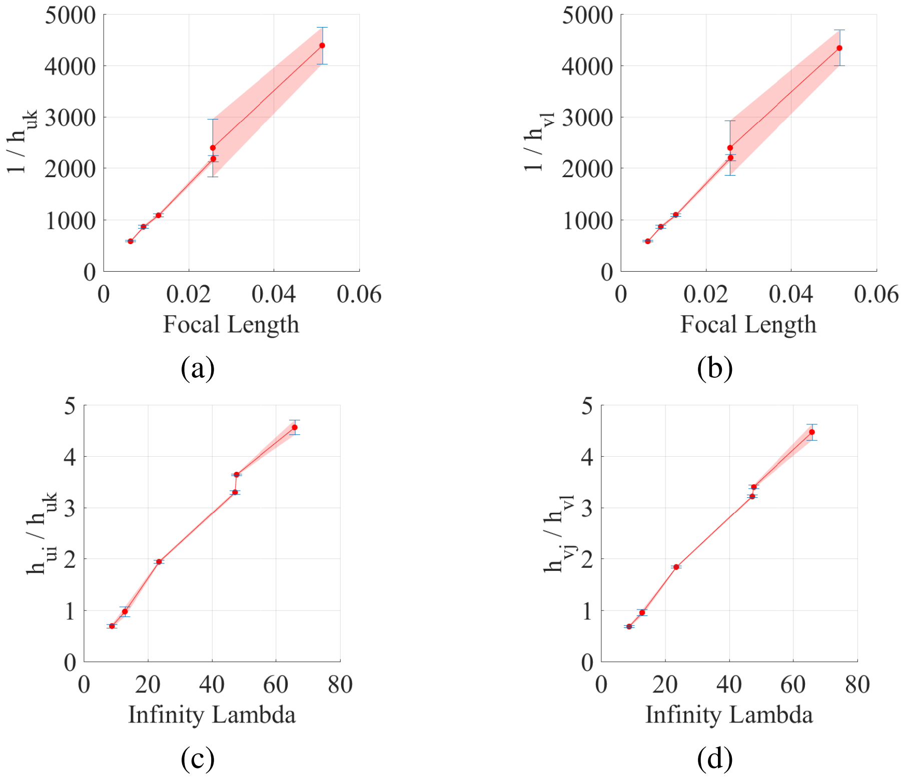


Figure 4.14: Relationships of the intrinsic matrix entries with the focal length and infinity lambda. The entries corresponding to the scale factors are represented against the focal length **(a)-(b)**. The entries corresponding to the principal point shifts are represented against the infinity lambda **(c)-(d)**.

has

$$\mathbf{H}\left(\vartheta\right)=\begin{bmatrix} \mathrm{a}\left(f\right) & 0 & 0 & 0 & 0 \\ 0 & \mathrm{b}\left(f\right) & 0 & 0 & 0 \\ \frac{\mathrm{e}(\lambda_\infty)}{\mathrm{c}(f)} & 0 & \frac{1}{\mathrm{c}(f)} & 0 & \frac{c_u}{\mathrm{c}(f)} \\ 0 & \frac{\mathrm{g}(\lambda_\infty)}{\mathrm{d}(f)} & 0 & \frac{1}{\mathrm{d}(f)} & \frac{c_v}{\mathrm{d}(f)} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.36}$$

where $\mathrm{a}\left(f\right)$, $\mathrm{b}\left(f\right)$, $\mathrm{c}\left(f\right)$, $\mathrm{d}\left(f\right)$, $\mathrm{e}\left(\lambda_\infty\right)$, and $\mathrm{g}\left(\lambda_\infty\right)$ define linear regression models from the observations of the metadata parameters.

4.4.3 Experimental Results

In this section, one employs the regression model (4.36) to describe the plenoptic camera for a specific zoom and focus settings.

The relationships $\mathrm{a}\left(f\right)$, $\mathrm{b}\left(f\right)$, $\mathrm{c}\left(f\right)$, $\mathrm{d}\left(f\right)$, $\mathrm{e}\left(\lambda_\infty\right)$, and $\mathrm{g}\left(\lambda_\infty\right)$, in (4.36), are estimated using the datasets in Table 4.10 except Dataset B. As in Section 4.4.2, one considered for each dataset 10 poses randomly selected from the acquired calibration poses to estimate the LFIM parameters and repeated this procedure 15 times to get the mean values. Notice that the LFIM parameters are mapped to the viewpoint camera array representation described in Section 4.2 and the mean values refer to the intrinsic matrix entries. Combining the intrinsic matrix entries with the observed focal length and infinity lambda, one can estimate the parameters of the affine mappings (Table 4.7) (step A of Figure 4.10).

Affine Mapping Parameters	$\mathrm{a}\,(f)$	$\mathrm{b}\,(f)$	$\mathrm{c}\,(f)$	$\mathrm{d}\,(f)$	$\mathrm{e}\,(\lambda_\infty)$	$\mathrm{g}\,(\lambda_\infty)$
Slope	35.1812	34.9393	85846.9190	84853.2935	0.0668	0.0655
y-Intercept	0.0281	0.0157	28.3403	48.7406	0.1793	0.1580

Table 4.7: Affine mapping parameters estimated for the relationships between the intrinsic matrix entries and the focal length or the infinity lambda identified in (4.36).

The Dataset B is not included in the previous analysis in order to be used to evaluate the accuracy of the camera representation (4.36) using the focal length and the infinity lambda meta-parameters. The camera representation is obtained by applying the affine mappings using the parameters identified in Table 4.7 for the focal length and infinity lambda of Dataset B (step B of Figure 4.10). These entries are compared with the mean values obtained by repeating 15 times the calibration procedure [41] using 10 randomly selected poses of Dataset B and are summarized in Table 4.8. The principal point $[c_u, c_v]^T$ is assumed to be the center of the VI since no relationship was found with the metadata parameters. Table 4.8 shows that the entries obtained from the calibration are similar to the ones obtained from the metadata. Namely, the maximum deviation is 7.8% and occurs for the principal point shift h_{ui}/h_{uk}.

Additionally, one considered a set of 10 randomly selected pattern poses of Dataset B to evaluate the re-projection, ray re-projection [41], and reconstruction errors using the LFIMs obtained from applying the calibration procedure [41] and from the regression model (4.36). The er-

Source	h_{si} **(mm)**	h_{tj} **(mm)**	$1/h_{uk}$	$1/h_{vl}$	h_{ui}/h_{uk}	h_{vj}/h_{vl}
From Calibration	0.3702	0.3281	858.6118	859.7984	3.6389	3.4056
From Metadata	0.3606	0.3459	839.5937	850.6042	3.3567	3.2778
Error (%)	2.6	5.4	2.2	1.1	7.8	3.8

Table 4.8: Intrinsic matrix entries estimated from focal length and infinity lambda using the affine mapping parameters in Table 4.7 and from calibration procedure [41] for Dataset B.

rors are summarized in Table 4.9. This table allows to have a more practical view of the difference between the two approaches considered. The errors presented are significant but is important to note that the extrinsic parameters are not explicitly estimated for the 10 poses considered. The re-projection and ray re-projection errors are similar, being greater for the representation obtained from the metadata by 0.34 pixels and 0.14 mm, respectively. On the other hand, the reconstruction error for the metadata based estimation is significantly greater than the one obtained from calibration [41] but still lower than 65 mm. However, note that the LFIM representation using the focal length and the infinity lambda is based on a statistical analysis between the metadata parameters provided by the camera manufacturer and the parameters estimated from a calibration procedure that are affected by noise.

Source	**Re-Projection Error (pixels)**	**Ray Re-Projection Error (mm)**	**Reconstruction Error (mm)**
From Calibration	5.7718	1.6172	10.0880
From Metadata	6.1162	1.7617	61.3519

Table 4.9: Re-projection, ray re-projection, and reconstruction errors associated with the estimation of the LFIM $\mathbf{H}$ from the regression model (4.36) and from the calibration procedure [41] for a set of 10 pattern poses of Dataset B.

4.5 Depth Capabilities

In an SPC, the multiple projections of a 3D point in a single exposure allow to recover the point's depth. In recent years, several works recover depth and shape from the LF using several cues [39, 102, 134, 139]. Nonetheless, references regarding the depth capabilities, *i.e.* the accuracy of the reconstructed depth, of an SPC for different zoom and focus settings are scarce.

In the literature, one can find works on the depth capabilities of Focused Plenoptic Cameras (FPCs) [77, 148] but not of SPCs. The most similar studies for SPCs correspond to the works of Hahne *et al.* [59, 61]. These studies estimate depth, depth of field and baselines for a simulated and a customized SPC using different optical parameters for the microlenses and for the main lens. These works require the specific knowledge of the parameters of the optical setup and are more focused on assisting the design of an SPC. Thus, in this section, is evaluated the depth capabilities of an SPC for a depth range between 0.05 and 2.00 m.

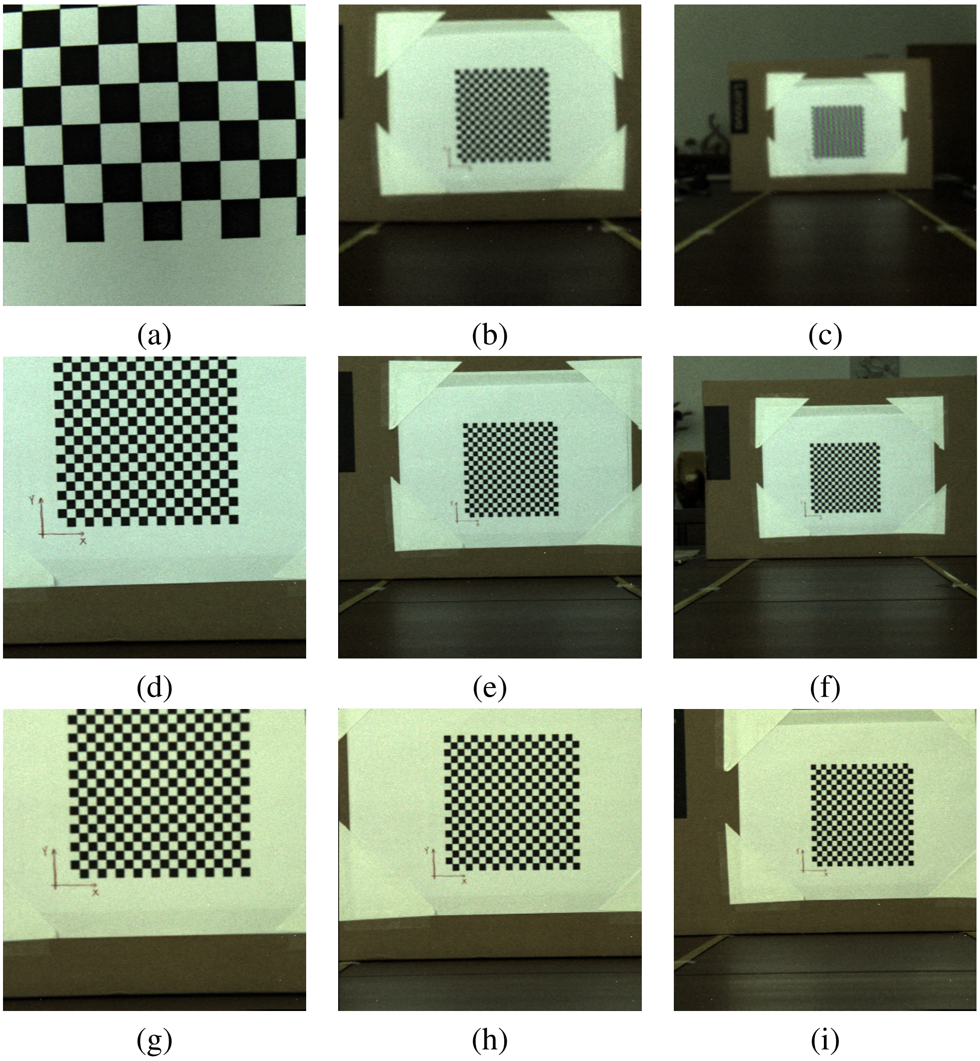
4.5.1 Datasets Acquisition and Calibration

The depth estimation using imagery of SPCs depends on the world plane in focus by the main lens. Hence, to study the depth capabilities, a combination of camera parameters must be analyzed to assess the reconstruction estimation accuracy (Section 4.4.1). More specifically, one acquired seven datasets under different zoom and focus settings [1] (Figure 4.15) using a 1^{st} generation Lytro camera. The zoom and focus settings of each dataset are determined by placing a target object at a pre-determined depth of the encasing of the camera and autofocusing on this object. This allows to define a plane in focus by the main lens that is close to the target object. Thus, the focus depth is assumed to be the depth of the target object.

The datasets acquired encompass images for calibration and depth range assessment. Namely, each dataset is provided with a set of calibration raw images since the camera parameters differ among datasets. Additionally, the calibration raw images are different from the depth raw images to ensure that the results do not suffer from any type of overfitting effect.

The calibration raw images are captured using a 19×19 calibration

[1] www.isr.tecnico.ulisboa.pt/∼nmonteiro/datasets/plenoptic/cviu2017/



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 4.15: VIs with 383×381 pixels for grid poses at different depths for Datasets A, D and F. The VIs for Dataset A correspond to grid poses at 0.05 m **(a)**, 0.55 m **(b)** and at 1.00 m **(c)**. The VIs for Dataset D correspond to grid poses at 0.55 m **(d)**, 1.10 m **(e)** and at 1.50 m **(f)**. The VIs for Dataset F correspond to grid poses at 1.10 m **(g)**, 1.50 m **(h)** and at 2.00 m **(i)**.

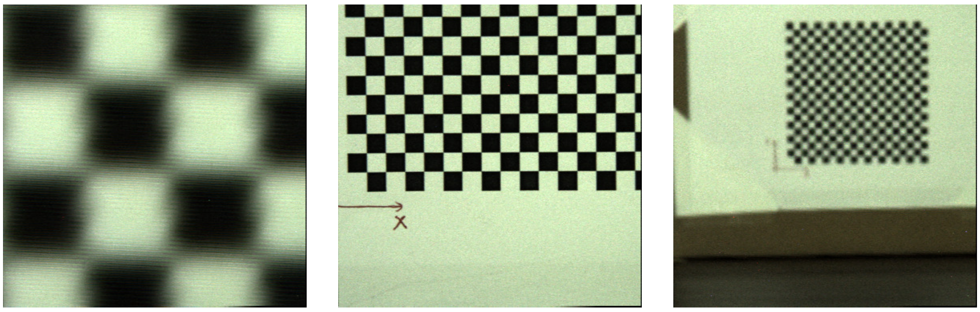
grid of 3.18 mm cells placed at different poses and at different depths near the target object depth bearing in mind that a minimum of 10 poses are required. On the other hand, the depth raw images are captured using two different grid sizes: 19×19 grid of 6.10×6.08 mm cells and 5×7 grid of 26.50×26.38 mm cells. The grids for the depth raw images are placed parallel to the encasing of the camera and at a regular spacing of 0.05 m from the camera for depth values ranging from 0.05 to 2.00 m. The two grid sizes are used for the depth raw images since the depth range evaluated is wide and it is necessary to have a reasonable number of detections to assess the depth accuracy. The smaller grid size is placed up to a maximum depth of 1.0, 1.5 and 2.0 m according to the focus depth considered 0.05, 0.50 and 1.50 m. The bigger grid size is placed considering all depth range evaluated. Table 4.10 summarizes the properties of the datasets acquired.

Dataset	Zoom Step*	Focus Step*	Focal Length*	Infinity Lambda*	Focus Depth (m)	Calibration Depth Range (m)	Calibration Poses	Ray Reprojection Error (mm)	Depth Poses
A	982	654	0.0064	23.5142	0.05	0.05 - 0.25	30	0.0993	60 (45)
B	754	941	0.0094	47.5966	0.05	0.05 - 0.35	30	0.1398	60 (37)
C	601	1212	0.0129	82.6425	0.05	0.10 - 0.40	14	0.2447	60 (29)
D	600	985	0.0130	8.7502	0.50	0.30 - 0.70	36	0.1357	70 (51)
E	335	1361	0.0258	47.2068	0.50	0.30 - 0.80	36	0.1267	70 (36)
F	337	1253	0.0256	12.8458	1.50	1.00 - 1.70	48	0.1806	80 (48)
G	100	1019	0.0513	65.9678	1.50	1.00 - 1.80	51	0.1381	80 (8)

Table 4.10: Information of the datasets acquired under different zoom and focus settings for a 1st generation Lytro camera. The meta-parameters are identified with the symbol *. The ray reprojection error [41] corresponds to the error obtained using the full set of calibration raw images. In the last column, the number of poses with detected features using the feature detector [78] is given within the parenthesis.

The depth ranges considered for the calibration poses are defined relatively to the plane in focus by the main lens and considering the FOV of the camera. Namely, the depth range is defined relatively to the target object depth to have sharper VIs which allow to detect more accurately the calibration grid points. The minimum depth value for the range is defined in order to have the full calibration grid in the VIs. In Figure 4.16, one can see the blurring that occurs for depths farther from the target object depth. On the other hand, the depth ranges used for the depth poses can be outside this range since the grids may fall out of the FOV.

The number of calibration raw images is different among the several datasets to ensure a ray reprojection error [41] below 0.2 mm for each dataset (Table 4.10). The only dataset that has a ray reprojection error higher than 0.2 mm is Dataset C. The maximum RMS for the ray reprojection error obtained during the calibration of the datasets is 0.2447 mm, which shows the accuracy of the calibration performed.



(a) Depth 0.05 m (b) Depth 0.50 m (c) Depth 1.50 m

Figure 4.16: VIs obtained from the depth raw images of the smaller grid of Dataset E. **(a)** VI for smaller grid placed at 0.05 m. **(b)** VI for smaller grid placed at 0.50 m. **(c)** VI for smaller grid placed at 1.50 m. The Dataset E has the world focal plane at 0.50 m, which leads to sharper images near the world focal plane **(b)** and blurred images as the pattern is farther from the world focal plane **(a)** and **(c)**.

4.5.2 Depth Range Assessment Datasets Preparation

The evaluation of the depth capabilities requires that the estimated points are given in the world coordinate system in order to be comparable with the ground truth points. Thus, one needs two steps: (i) detect the projections of a given grid point and associate the projections with that 3D point, and (ii) change the coordinate system associated with the estimated points.

Feature Detection and Correspondences. For the evaluation of the reconstruction estimation accuracy, besides the LFIM one also needs to know the imaged points obtained for each pose of the grids captured in the depth raw images. These imaged points are the projections of each of the grid points captured by the camera. The corners are detected by applying a feature detector [78] to each of the VIs obtained from the raw image after the decoding process [41]. This is similar to the feature detection used during the calibration procedure. The major difference is that, for the depth raw images, the grids may fall out of the FOV and, therefore, the number of imaged points is not constant throughout all grid poses.

Although many grid poses are acquired for assessing the depth range of these cameras, the feature detection procedure discards many of these poses since there are no identifiable features (Table 4.10). Furthermore, for the depth range considered, some of the datasets only have features for a few number of depth values. The process of feature detection makes Dataset C unusable for the smaller grid size, and the Dataset G unusable for both grid sizes used. The correspondences are obtained by grouping the projection rays obtained from each VI that correspond to the same grid point in the object space.

Camera to World Coordinate System Transformation. The LFIM and the set of projection rays associated to a given grid point in the object space allows to recover the point's depth in the camera coordinate system (Section 7.2). Thus, one needs to estimate the rigid body transformations between the world and the camera coordinate systems defined for each of the datasets. For each dataset, the transformation is estimated using a Procrustes analysis [79] between the estimated points in the camera coordinate system and the ground truth points in the world coordinate system.

The grids captured for each set of depth raw images are only moved along the z-axis forming a parallelepiped. This allows to easily obtain the ground truth points in the world coordinate system. On the other hand, the estimated points for the grid points detected in the depth raw images do not form a parallelepiped due to noise and to the reconstruction capabilities of the camera. Nonetheless, the grid points associated to a given grid pose form a planar surface that is present in both coordinate systems. Hence, one can use this knowledge to remove the estimated points associated with grid depths that deviate from a planar surface. The estimated points discarded from the Procrustes analysis correspond to grid depths whose fitting error to a planar surface is above a given threshold. This threshold is defined as the mean of the planar fitting errors for all grid depths in the depth raw images for a given dataset.

Figure 4.17 shows the result of applying the estimated rigid body transformations to convert the estimated points from the camera coor-

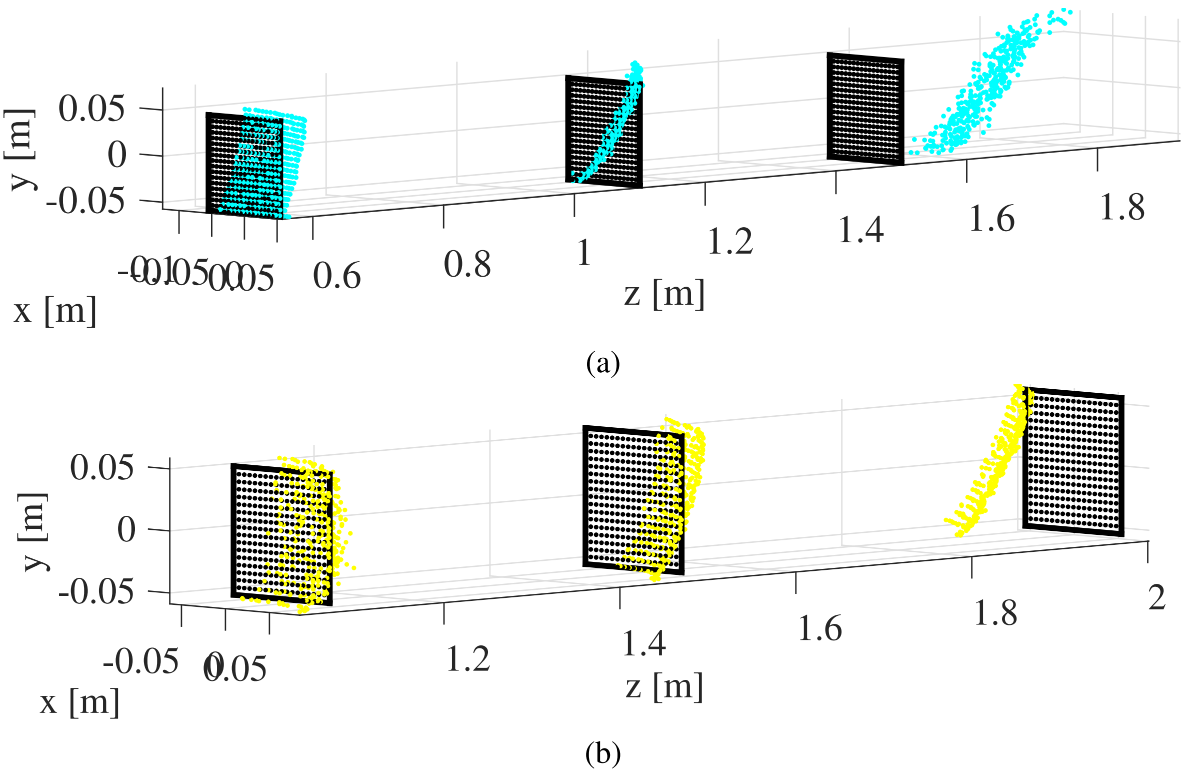


Figure 4.17: The estimated (cyan and yellow) and ground truth (black) grid points obtained for Datasets D and F using a smaller grid are depicted in **(a)** and **(b)**. The planar surfaces correspond to grids at the depth limits of the Datasets D and F and at an intermediate depth value (0.55 m, 1.10 m and 1.50 m for Dataset D, and 1.10 m, 1.50 m and 2.00 m for Dataset F).

dinate system to the world coordinate system for three depth values of Datasets D and F using the smaller grid. Although the estimated points do not lie in a plane (reconstruction is done on a point by point basis), one can see that the estimated grids are close to planar surfaces. Additionally, comparing the grid at depth 1.5 m in each of the datasets, one can see that the estimated points for Dataset F define more accurately a planar surface (RMS error of 6.6 mm and 3.0 mm for Datasets D and F, respectively) that is close to the corresponding ground truth grid (RMS error of 0.2393 m and 0.0448 m for Datasets D and F, respectively). Thus, is expected that increasing the zoom and focus depth will originate better estimates for points farther from the camera.

4.5.3 Reconstruction Estimation Accuracy

Using the correspondences, the LFIM and the rigid body transformations, one obtains an estimate for the grid points in the world coordinate system that can be used to compute the mean reconstruction error

$$r_e = \frac{1}{P} \sum_{i=1}^{P} \|\mathbf{m}_i - \hat{\mathbf{m}}_i\| \tag{4.37}$$

where P is the number of points, $\mathbf{m}_i$ is the ground truth point and $\hat{\mathbf{m}}_i$ is the estimated point. The reconstruction errors and the estimated depth for the datasets are depicted in Figures 4.18 and 4.19. The depth ranges identified for each of the datasets as well as the mean and STD for the normalized reconstruction errors are summarized in Table 4.11. The normalized reconstruction errors are obtained by dividing the reconstruction errors by the corresponding ground truth depths. The depth ranges are identified by determining the regions where the mean of the normalized reconstruction errors is lower or equal to 10%.

Dataset	Depth Range (m)	Mean ± STD Error in Depth Range (%)	Mean ± STD Error (%)
A	0.35 - 1.30	6.74 ± 5.13	16.67 ± 6.18
B	0.40 - 1.30	7.89 ± 5.96	13.72 ± 9.73
C	0.05 - 0.05	1.34 ± 5.93	25.73 ± 18.12
D	0.60 - 2.00	5.13 ± 3.20	14.01 ± 5.00
E	0.75 - 2.00	5.44 ± 3.30	8.28 ± 4.19
F	0.90 - 2.00	3.68 ± 1.78	5.90 ± 2.03
G	1.50 - 1.85	1.93 ± 0.60	1.93 ± 0.60

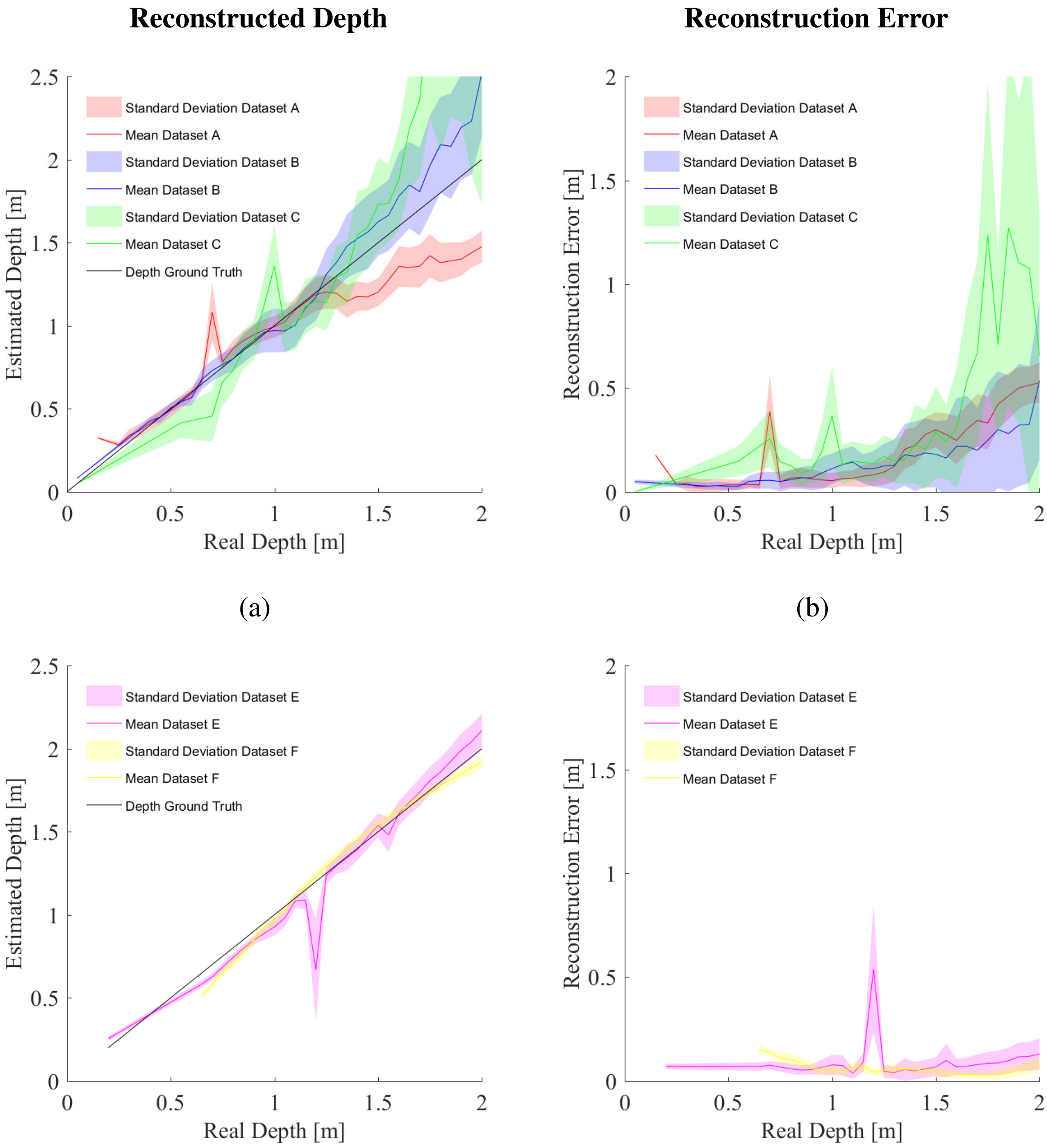
Table 4.11: Depth ranges for the datasets acquired. The depth ranges are identified as the regions whose mean for the normalized reconstruction errors is lower or equal to 10%. The mean and STD for the normalized reconstruction errors within the depth ranges defined and for all ground truth depths are also depicted.

Zoom Step Analysis. In Figure 4.18.a-b, the datasets are grouped by constant focus depths. Namely, the figure conveys information of datasets with focus depth at 0.05 m. For this focus depth, is possible to see that the mean reconstruction error for points farther from the plane in focus is higher. This is also highlighted by the difference of the normalized error in the depth range and in the overall depth analyzed presented in Table 4.11. Additionally, is observed that the increase in zoom allows to have a lower reconstruction error for points farther from the focus depth. Table 4.11 shows the normalized error in the whole depth analyzed decreases while the normalized error in the depth range is maintained when the zoom is increased (excluding Dataset C due to the unusually high normalized reconstruction errors).

Focus Depth Analysis. In Figure 4.18.c-d, the datasets are grouped by similar zoom step (focal length). Namely, the figure conveys information of datasets with zoom step close to 336. For this zoom step, the focus depth appears to improve the reconstruction error for points at depths near the focal plane. In Table 4.11, this is highlighted by the change of the depth range that has a normalized reconstruction error lower or equal to 10%.

Zoom Step and Focus Depth Analysis. In Figure 4.19, Datasets A, D and F are depicted to highlight the reconstruction error by modifying both zoom and focus settings. This figure shows the reconstruction error decreasing as the zoom step increases and the depth for which there are features detected also change. This can also be seen by the decrease on the normalized error for the whole depth analyzed and by the shift on the depth range with normalized error lower or equal to 10% in Table 4.11.

The lower reconstruction error with increasing zoom can be explained by considering the depth error ε_z of a binocular stereo configuration $\varepsilon_z = \frac{z^2}{b\,f}\varepsilon_d$, where b is the baseline length, f is the focal length, z is the depth of a point in the object space, and ε_d is the disparity error. The increase in zoom corresponds to an increase in the focal length f which leads to a decrease on the depth error, which is in accordance with the findings in this figure. On the other hand, the focus depth determines the depth at which the minimum reconstruction error occurs and, implicitly, the depth range. This can be explained looking at the ray-spaces.



(a)

(b)

(c)

(d)

Figure 4.18: Reconstruction estimation accuracy with zoom step **(first row)** and with focus depth **(second row)**. The **first column** depicts the reconstructed depth while the **second column** depicts the reconstruction error for the estimated points obtained for datasets A through E. The **first row** groups the datasets with focus depth at 0.05 m (Datasets A, B and C) and the **second row** groups the datasets with zoom step close to 336 (Datasets E and F).

Namely, a point in the world focal plane corresponds to a vertical line in this space, which leads to a smaller error due to a smaller discretization error (smaller ε_d) that occurs at the image sensor (staircase effect). As the point moves away from the world focal plane, the line starts to de-

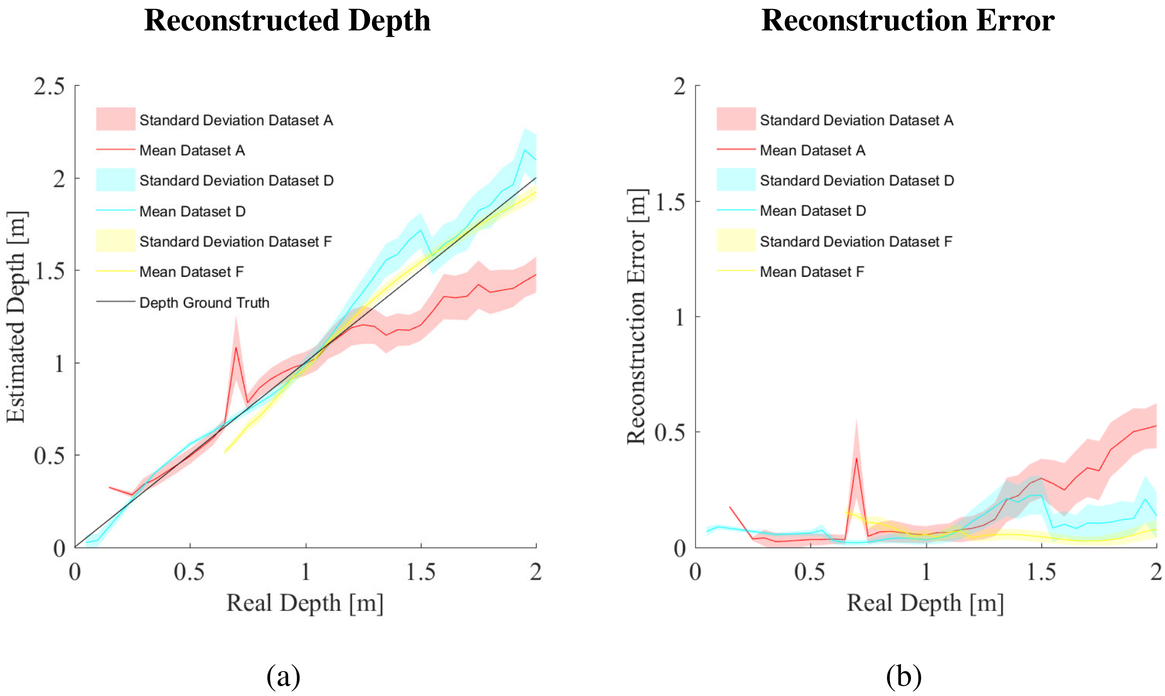


Figure 4.19: Reconstruction estimation accuracy with zoom step and focus depth. The **first column** depicts the reconstructed depth while the **second column** depicts the reconstruction error for the estimated points obtained for datasets with different zoom and focus settings.

viate from this vertical line and the discretization error increases (higher ε_d) leading to an increase on the reconstruction error. Notice that the reconstruction method presented in Section 7.2.1 reduces but does not eliminate the reconstruction error associated with discretization.

The reconstruction results presented in this section are obtained considering the radial distortion parameters. The mean difference of the estimated points normalized by the ground truth depth not considering radial distortion parameters is less than 1.6% for all datasets analyzed (Appendix D). This difference does not change significantly the results presented in Table 4.11. Thus, it is considered that the radial distortion does not play an important role on the reconstruction estimation accuracy.

In summary, the results presented show that SPCs have a reconstruction estimation accuracy that varies with the zoom and focus settings of the camera. The zoom is a determinant factor on increasing the reconstruction accuracy of these cameras, while the focus depth (as a combination of zoom and focus steps) plays a role on shifting the depth range. The depth range analyzed from 0.05 m to 2.00 m can be reconstructed with accuracy by choosing correctly the zoom and focus settings of the camera.

4.6 Chapter Summary

In this chapter, was defined the mapping between the LFIM used to describe an SPC and the viewpoint cameras, and was established the equivalence between the projection set $\mathcal{P}_{kl}$ (3.16) and the projections of the viewpoint cameras. The viewpoint cameras define a coplanar array of cameras that differ on the location of their projection centers and on their principal points (Section 4.1.2). The different principal points define an EPI geometry whose zero disparity plane is at a finite depth, the main lens world focal plane (Section 4.1.3). The EPI geometry (4.15) extends the geometry defined by Bolles *et al.* [22] that considers images acquired by identical cameras, *i.e.* same principal point.

The pinhole viewpoint camera constraint (4.11) allows to represent the LFIM introduced in [41] using eight free intrinsic parameters. This is accomplished by shifting the rays parameterization plane along the optical axis of the camera [17] to the plane containing the viewpoint projection centers and removing redundant parameters with the extrinsic parameters (Section 4.2).

The viewpoint camera array model derived in Section 4.1 is used to define a linear solution for the SPC that considers two steps: i) a DLT calibration to obtain the parameters that describe the viewpoint parametric homography matrix from point correspondences $(\tilde{\mathbf{m}}, \tilde{\mathbf{q}})$, (ii) and a strategy to decompose this homography matrix into intrinsic and extrinsic parameters based on a parametric representation of the image of the absolute conic. This is the first work capable of estimating the principal point shift in the linear solution (Section 4.3) which allows to outperform state of the art methods.

Additionally, one shows that an SPC can be represented by the regression model (4.36) based on the observations of the main lens focal length and infinity lambda meta-parameters provided by the camera manufacturer with the images acquired. This allows to propose a calibration scheme for an SPC without having to acquire a new calibration dataset for a specific zoom and focus settings (Section 4.4).

The depth capabilities of an SPC are evaluated for depths ranging from 0.05 to 2.00 meters. The experimental findings suggest that these cameras are capable of reconstructing points in the depth range analyzed by appropriately choosing the zoom and focus settings. Namely, the zoom increase allows to lower the reconstruction error while the focus depth determines the depth range of the camera. This is the first work that studies the depth capabilities of an SPC.

The next chapter defines the geometry of the microlens array and exploits this geometry to propose a calibration procedure for a Multifocus Plenoptic Camera (MPC).

Chapter 5

Multifocus Plenoptic Camera

A Multifocus Plenoptic Camera (MPC) images a world point at multiple sensor locations due to the array of microlenses behind the main lens. The multiple focal lengths on the array imply that the same scene point is imaged in each microlens type with different degrees of defocus (Figure 5.1). However, an MPC allows 3D reconstruction from a single image, provided the camera is accurately calibrated.

The geometry of an MPC is based on the geometry of a Focused Plenoptic Camera (FPC) [91, 120] that generates focused Microlens Images (MIs) by placing the focal plane of the microlenses on the main lens focal plane. Hence, this chapter builds from the model of Dansereau *et al*. [41] and derives the mapping between the Lightfield Intrinsic Matrix (LFIM) and microlens camera array that allows to fully formalize the projection model for a microlens camera. Using the geometry of the microlens array complemented with a blur model associated with each microlens type, it is proposed a calibration approach for an MPC.

5.1 Microlens Camera Array

The LFIM can represent an array of distinct coplanar and parallel cameras (Section 4.1). In Section 4.1, one presented the mapping of the LFIM to a viewpoint array that describes a Standard Plenoptic Camera (SPC). In this section, is shown that an FPC can be modeled by a microlens array which can be obtained from the LFIM [21, 149].

The MI (Figure 5.1.a) results from the rays that cross the center of a specific microlens. In this case, the coordinates (k, l) are the indices associated with each MI and the coordinates (i, j) encode the position of a pixel in the MI. Let the projection matrix $\mathbf{P}^{kl}$ describe a microlens camera parameterized by the coordinates $(k, l) \in \mathbb{Z}^2$

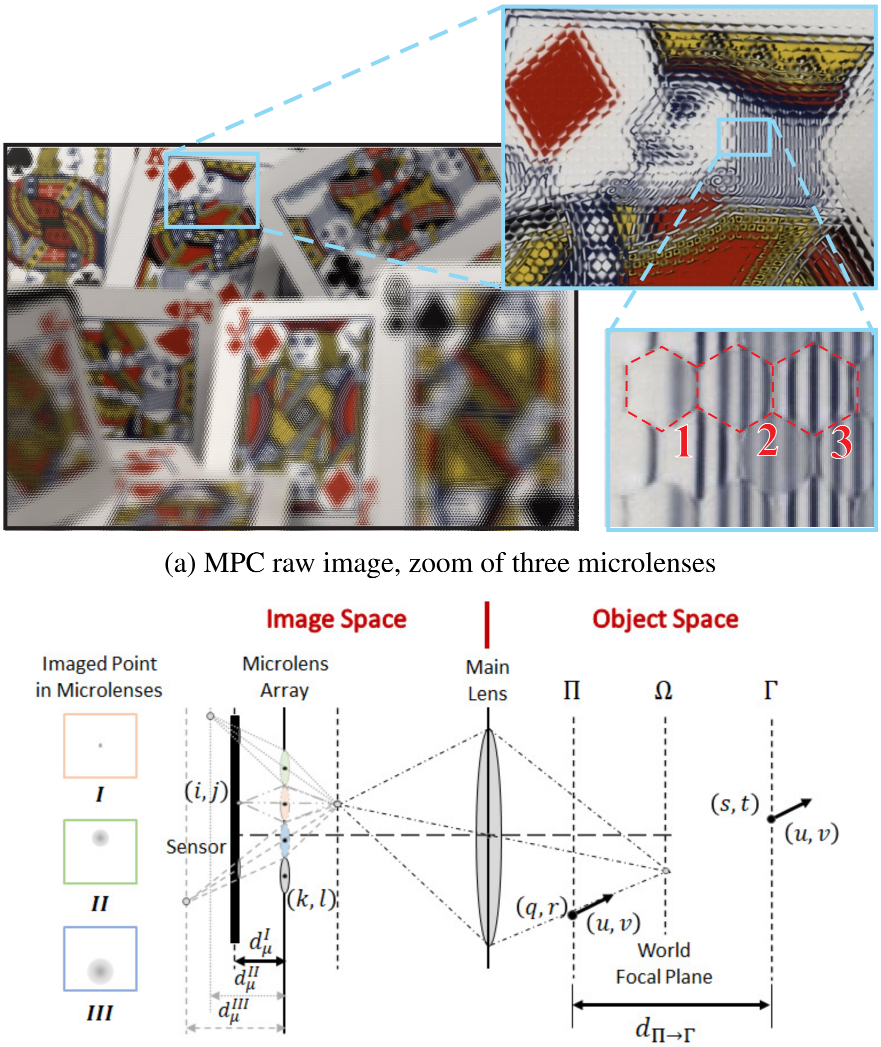
$$\mathbf{P}^{kl} = \mathbf{K}^{kl} \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{t}^{kl} \end{bmatrix} {}^{c}\mathbf{T}_{w} \tag{5.1}$$

where $\mathbf{K}^{kl}$ denotes the intrinsic matrix, $\mathbf{I}_{3\times3}$ is a 3×3 identity matrix, $\mathbf{t}^{kl}$ is the projection center and ${}^{c}\mathbf{T}_{w} = \begin{bmatrix} {}^{c}\mathbf{R}_{w} & {}^{c}\mathbf{t}_{w} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}$ defines the rigid body transformation between the world and camera coordinate systems with rotation ${}^{c}\mathbf{R}_{w} \in SO(3)$ and translation ${}^{c}\mathbf{t}_{w} \in \mathbb{R}^3$, and $\mathbf{0}_{1\times3}$ is the 1×3 null matrix.

Similarly to the viewpoint array, ${}^{c}\mathbf{T}_{w}$ defines one coordinate system for all microlens cameras while the intrinsic matrix and the projection center are different for each microlens camera (k, l). In the following, let the camera model for the microlens array (5.1) take into account that the principal point and the projection center are different for each camera while the scale factor remains the same:

$$\mathbf{K}^{kl} = \begin{bmatrix} k_u & 0 & u_0 + k\,\Delta u_0 \\ 0 & k_v & v_0 + l\,\Delta v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{t}^{kl} = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} + \begin{bmatrix} k\,\Delta x_0 \\ l\,\Delta y_0 \\ 0 \end{bmatrix} \tag{5.2}$$

where the scalars k_u and k_v denote scale factors, the vector $[u_0, v_0]^T$ defines the principal point for the microlens camera $(k, l) = (0, 0)$, and the vectors $[\Delta u_0, \Delta v_0]^T$ and $[\Delta x_0, \Delta y_0, 0]^T$ denote principal point shift and baseline between consecutive microlens cameras, respectively. The vector $[x_0, y_0, z_0]^T$ defines the location of the microlens camera array relatively to the camera coordinate system origin. This allows to represent the microlens camera array using a maximum of 11 parameters.



(a) MPC raw image, zoom of three microlenses

(b) MPC geometry, microlenses with three focal lengths

Figure 5.1: Multifocus effect. **(a)** Image acquired by an MPC [5]. Small region is augmented to show microlens borders and focusing. MIs, 1 and 2 are blurred, 3 is focused. **(b)** MPC geometry illustrating the focused and blurred image formation.

5.1.1 Projection Model

Considering a parametric representation for the the projection matrix of a microlens camera (k, l), similar to the one presented for a viewpoint camera (i, j) (4.3), and following the same steps defined in Section 4.1.1, one defines the projection of a point $\mathbf{m} = [x, y, z]^T$ to a point in the image plane $\tilde{\mathbf{q}} = [i, j, 1]^T$ of a particular camera (k, l) as

$$\tilde{\mathbf{q}} \sim \underbrace{\begin{bmatrix}\mathbf{I}_{3\times3} & k\,\mathbf{I}_{3\times3} & l\,\mathbf{I}_{3\times3}\end{bmatrix}}_{\mathbf{M}} \begin{bmatrix}\mathbf{P}^0 \\ \mathbf{\Delta P}^k \\ \mathbf{\Delta P}^l\end{bmatrix} \tilde{\mathbf{m}} \quad , \tag{5.3}$$

where the symbol $\sim$ denotes equal up to a scale factor with

$$\begin{aligned}
\mathbf{P}^0 &= \begin{bmatrix} k_u & 0 & u_0 \\ 0 & k_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} & x_0 \\ \mathbf{I}_{3\times3} & y_0 \\ & z_0 \end{bmatrix} {}^c\mathbf{T}_w \quad , \\
\mathbf{\Delta P}^k &= \begin{bmatrix} \mathbf{0}_{3\times2} & \begin{array}{cc} \Delta u_0 & k_u\,\Delta x_0 + \Delta u_0\, z_0 \\ \multicolumn{2}{c}{\mathbf{0}_{2\times2}} \end{array} \end{bmatrix} {}^c\mathbf{T}_w \quad \text{and} \\
\mathbf{\Delta P}^l &= \begin{bmatrix} \mathbf{0}_{3\times2} & \begin{array}{cc} \multicolumn{2}{c}{\mathbf{0}_{1\times2}} \\ \Delta v_0 & k_v\,\Delta y_0 + \Delta v_0\, z_0 \\ \multicolumn{2}{c}{\mathbf{0}_{1\times2}} \end{array} \end{bmatrix} {}^c\mathbf{T}_w \quad .
\end{aligned} \tag{5.4}$$

The matrix $\mathbf{M}$ provides an easy way to add the several camera indices available for a plenoptic camera and in this way get the multiple projections for a point $\mathbf{m}$ in the object space.

The projection (5.3) using the microlens coordinates (k, l) is equivalent to the projection set $\mathcal{P}_{ij}$ defined in Section 3.4.1.

5.1.2 Mapping from LFIM to Microlens Projection Matrices

In order to obtain the mapping from the LFIM to the camera model (5.1) let us first define the projection centers of the microlens cameras, and then define the projection equation considering the LFIM $\mathbf{H}$ and (k, l) as parameters.

Microlens Projection Centers. Following the same steps described in Section 4.1.2 to compute the caustic surface using the Jacobian method [27], one obtains two solutions for λ solving the vanishing constraint (see Appendix E for details):

$$\lambda_1 = -\frac{h_{qi}}{h_{ui}} \quad \vee \quad \lambda_2 = -\frac{h_{rj}}{h_{vj}} \quad . \tag{5.5}$$

Notice that for a microlens camera, the microlens coordinates (k, l) are fixed and are considered as parameters. This allows to define the positions (q, r) and the directions (u, v) as affine mappings only on the pixel coordinates (i, j).

The solutions to the vanishing constraint allow to identify the caustic profile for a single microlens camera. More specifically, the caustic profile consists of a line with (i) unique (x, z) and variable y components if $\lambda = \lambda_1$ or (ii) unique (y, z) and variable x components if $\lambda = \lambda_2$. In case $\lambda_1 \neq \lambda_2$ the microlens is a non-central camera. The microlens camera corresponds to a central camera, *i.e.* a camera with a unique projection center, if and only if $\lambda_1 = \lambda_2$ which imply the model parameters relation

$$\frac{h_{qi}}{h_{ui}} = \frac{h_{rj}}{h_{vj}} \quad . \tag{5.6}$$

Assuming this constraint, the location of the microlens projection center for a microlens camera (k, l) is given by

$$\mathbf{p}_c = \begin{bmatrix} h_q - \frac{h_{qi}}{h_{ui}} h_u + k\left(h_{qk} - \frac{h_{qi}}{h_{ui}} h_{uk}\right) \\ h_r - \frac{h_{rj}}{h_{vj}} h_v + l\left(h_{rl} - \frac{h_{rj}}{h_{vj}} h_{vl}\right) \\ -\frac{h_{qi}}{h_{ui}} \end{bmatrix} \quad . \tag{5.7}$$

Furthermore, considering all microlens cameras that can be defined, the LFIM represents a coplanar grid of equally spaced projection centers. Notice that the microlens coordinates (k, l) only affect the x- and y-components of the projection centers while the z-component of the projections centers is always the same.

LFIM Mapping. Considering that the rays of one microlens camera converge to a unique point (5.6), one may set constant the values (k, l) and solve (3.11) relatively to (i, j). This gives an equation of a microlens pixel (i, j) imaging the 3D point (x, y, z) that can be rewritten as a pinhole model like (5.1) with the intrinsic matrix defined as

$$\mathbf{K}^{kl} = \begin{bmatrix} \frac{1}{h_{ui}} & 0 & -\frac{h_u}{h_{ui}} - k\,\frac{h_{uk}}{h_{ui}} \\ 0 & \frac{1}{h_{vj}} & -\frac{h_v}{h_{vj}} - l\,\frac{h_{vl}}{h_{vj}} \\ 0 & 0 & 1 \end{bmatrix} \quad , \tag{5.8}$$

and the projection center as $\mathbf{t}^{kl} = -\mathbf{p}_c$ (5.7). This allows to obtain the mappings to the representations in (5.2). Namely, comparing (5.8) with (5.2), one identifies a common component $[u_0, v_0]^T = -\left[h_u/h_{ui}, h_v/h_{vj}\right]^T$ and a differential (shift) component $[\Delta u_0, \Delta v_0]^T = -\left[h_{uk}/h_{ui}, h_{vl}/h_{vj}\right]^T$ on the principal point. The scale factors are defined as $k_u = 1/h_{ui}$ and $k_v = 1/h_{vj}$, and the baseline is defined as $[\Delta x_0, \Delta y_0, 0]^T = -\left[h_{qk} - h_{uk}\, h_{qi}/h_{ui}\right.$ The position of the microlens camera array relatively to the camera coordinate system origin is defined as $[x_0, y_0, z_0]^T = -\left[h_q - h_u\, h_{qi}/h_{ui},\ h_r - h_v\, h_{rj}\right.$

5.1.3 Properties of Microlens Projection Matrices

Considering equation (3.11), one can obtain the Epipolar Plane Image (EPI) geometry that relates the depth of a point with the disparity on the MIs $\left[\frac{\Delta i}{\Delta k}, \frac{\Delta j}{\Delta l}\right]^T$

$$\frac{\Delta i}{\Delta k} = -\frac{h_{qk} - \frac{h_{qi}}{h_{ui}} h_{uk}}{h_{ui}} \frac{1}{z + \frac{h_{qi}}{h_{ui}}} - \frac{h_{uk}}{h_{ui}} \text{ and } \frac{\Delta j}{\Delta l} = -\frac{h_{rl} - \frac{h_{rj}}{h_{vj}} h_{vl}}{h_{vj}} \frac{1}{z + \frac{h_{rj}}{h_{vj}}} - \frac{h_{vl}}{h_{vj}} \,. \tag{5.9}$$

The mapping (5.7) and (5.8) allows to rewrite the EPI geometry defined in equation (5.9) as

$$\frac{\Delta i}{\Delta k} = k_u \frac{\Delta x_0}{z + z_0} + \Delta u_0 \quad \text{and} \quad \frac{\Delta j}{\Delta l} = k_v \frac{\Delta y_0}{z + z_0} + \Delta v_0 \,. \tag{5.10}$$

The EPI geometry shows that the zero disparity plane for the microlens cameras is at the plane containing the viewpoint projection centers $z_v = -\frac{h_{qk}}{h_{uk}} = -\frac{h_{rl}}{h_{vl}}$. This is in accordance with the singularities described for the microlens cameras in Section 3.4.2.

5.2 Reducing the Parameters of the LFIM

The LFIM has 12 non-zero entries (4.7) but some parameters can be avoided by considering them on the extrinsic parameters and choosing an appropriate camera coordinate system origin. Namely, choosing the camera coordinate system origin at the plane containing the microlens projection centers.

Repeating the same steps defined in Section 4.2 and assuming that the plane Γ corresponds to the plane containing the microlens projection centers at $d_{\Pi\to\Gamma} = -h_{qi}/h_{ui}$, one obtains $[x_\Gamma, y_\Gamma, z_\Gamma]^T = [h_{sk}\ k, h_{tl}\ l, 0]^T + \lambda\ [u, v, 1]^T$. with $x_\Gamma = x - h_s$, $y_\Gamma = y - h_t$ and $z_\Gamma = z - d_{\Pi\to\Gamma}$. In this case, the LFIM $\mathbf{H}_\Gamma$ with 8 non-zero entries is given by

$$\mathbf{H}_\Gamma = \begin{bmatrix} 0 & 0 & h_{sk} & 0 & 0 \\ 0 & 0 & 0 & h_{tl} & 0 \\ h_{ui} & 0 & h_{uk} & 0 & h_u \\ 0 & h_{vj} & 0 & h_{vl} & h_v \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \,. \tag{5.11}$$

Considering this representation for the LFIM, the microlens projection centers location (5.7) reduces to $\mathbf{p}_c = [k\ h_{sk}, l\ h_{tl}, 0]^T$.

The LFIMs (4.17) and (5.11) map rays in the image space to rays in the object space defined by a point and a direction considering the camera coordinate system origin either at the plane containing the viewpoint or the microlens projection centers, respectively. Changing the parameterization of the rays in the object space for their intersection with two planes using (3.5), one can further reduce the number of parameters used to define the LFIM. More specifically, considering that the two planes correspond to the plane Γ containing the viewpoint projection centers ((3.3) with $d_{\Pi\to\Gamma} = -h_{qk}/h_{uk}$) and the plane Θ containing the microlens projection centers at a distance $d_{\Gamma\to\Theta} = -h_{si}/h_{ui}$ (3.5), one obtains a LFIM with 6 non-zero entries

$$\mathbf{H}_{\Gamma,\Theta} = \begin{bmatrix} h_{si} & 0 & 0 & 0 & 0 \\ 0 & h_{tj} & 0 & 0 & 0 \\ 0 & 0 & d_{\Gamma\to\Theta}\, h_{uk} & 0 & d_{\Gamma\to\Theta}\, h_u \\ 0 & 0 & 0 & d_{\Gamma\to\Theta}\, h_{vl} & d_{\Gamma\to\Theta}\, h_v \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} . \tag{5.12}$$

5.3 Reducing the Parameters of the Microlens Array

The representation of the LFIM considering the plane with the microlenses projection centers (5.11) allows to represent the microlens array using 8 parameters. Nonetheless, the microlens array can be represented using fewer parameters.

Considering that the FPC is described by the LFIM with 8 non-zero parameters and with the parameterization plane corresponding to the plane containing the viewpoint projection centers (4.17), one obtains a pinhole model for the microlens camera (k, l) with the intrinsic matrix (5.8) and with the projection center defined as

$$\mathbf{t}^{kl} = \begin{bmatrix} \frac{h_u}{h_{ui}} h_{si} \\ \frac{h_v}{h_{vj}} h_{tj} \\ \frac{h_{si}}{h_{ui}} \end{bmatrix} + \begin{bmatrix} k\, \frac{h_{uk}}{h_{ui}} h_{si} \\ l\, \frac{h_{vl}}{h_{vj}} h_{tj} \\ 0 \end{bmatrix} . \tag{5.13}$$

Comparing with (5.2), the baseline is defined as $[\Delta x_0, \Delta y_0, 0]^T = [h_{si}\, h_{uk}/h_{ui},$

$h_{tj}\, h_{vl}/h_{vj},\ 0]^T$ and the vector $[x_0, y_0, z_0]^T = \left[h_{si}\, h_u/h_{ui}, h_{tj}\, h_v/h_{vj}, h_{si}/h_{ui}\right]^T$.

The mapping with the entries of the LFIM allows to redefine the translation vector $\mathbf{t}^{kl}$ using the intrinsic parameters in (5.2) as

$$\mathbf{t}^{kl} = \begin{bmatrix} u_0\, b_x \\ v_0\, b_y \\ -k_u\, b_x \end{bmatrix} + \begin{bmatrix} k\, \Delta u_0\, b_x \\ l\, \Delta v_0\, b_y \\ 0 \end{bmatrix} \quad , \tag{5.14}$$

where $\left[b_x, b_y, 0\right]^T = \left[-h_{si}, -h_{tj}, 0\right]^T$ corresponds to the baseline between consecutive viewpoint cameras [104] (Section 4.2). This allows to represent the microlens camera array using 8 parameters. Furthermore, considering the microlens pinhole constraint $k_u\, b_x = k_v\, b_y$ (5.6), the microlens camera array can be defined using 7 parameters.

Finally, let us consider the EPI geometry (4.15) defined using the LFIM $\mathbf{H}_\Gamma$ (4.17) as

$$\frac{\Delta k}{\Delta i} = -\frac{h_{si}}{h_{uk}} \frac{1}{z_{ik}} - \frac{h_{ui}}{h_{uk}} \quad \text{and} \quad \frac{\Delta l}{\Delta j} = -\frac{h_{tj}}{h_{vl}} \frac{1}{z_{jl}} - \frac{h_{vj}}{h_{vl}} \tag{5.15}$$

where the z-coordinate of the point in the object space is denoted by z_{ik} and z_{jl}. Notice that the pair of coordinates (i, k) and (j, l) are assumed to be independent, so there is no guarantee that the depth of the point resulting from the EPI geometry will be the same. A point will have an unique depth if one considers the following constraint

$$\frac{h_{uk}}{h_{si}} \frac{\Delta k}{\Delta i} = \frac{h_{vl}}{h_{tj}} \frac{\Delta l}{\Delta j} \tag{5.16}$$

assuming that (5.6) is also satisfied. The constraint (5.16) is not directly applicable to the entries of the LFIM since one should have a set of corresponding rays in the image space that are associated with the same point in the object space to compute the disparities $\frac{\Delta k}{\Delta i}$ and $\frac{\Delta l}{\Delta j}$. However, assuming that the disparities on the Viewpoint Images (VIs) are equal, *i.e.* $\frac{\Delta k}{\Delta i} = \frac{\Delta l}{\Delta j}$, one can redefine (5.16) as

$$\frac{h_{uk}}{h_{si}} = \frac{h_{vl}}{h_{tj}} \quad . \tag{5.17}$$

This allows to apply the constraint directly to the entries of the LFIM without needing extra information. This constraint is similar to the one defined in Zhang *et al.* [150].

Representing the restriction using the intrinsic parameters in (5.2), one has $\frac{\Delta u_0}{k_u \; b_x} = \frac{\Delta v_0}{k_v \; b_y}$ which can be simplified to

$$\Delta u_0 = \Delta v_0 \tag{5.18}$$

considering the microlens pinhole constraint (5.6). This allows to represent the microlens camera array using a minimum of 6 parameters: 5 parameters to represent the intrinsic matrix $\mathbf{K}^{kl}$ and 1 parameter to represent the translation vector $\mathbf{t}^{kl}$.

5.4 Bok *et al.* Mapping to LFIM

In Section 5.1, one established the mapping between the LFIM and a microlens camera projection model. In this section, one shows that the model proposed by Bok *et al.* [21] can be represented by a projection matrix, and consequently a LFIM, similar to the one in (4.17), constraining the microlenses centers coordinates on the raw image to be regularly spaced. The structure of the projection matrix and LFIM depends on the sampling basis that is considered for representing the microlenses coordinates (Section 3.5).

The microlens camera model, defined by Bok *et al.* [21] and adapted by Nousias *et al.* [115] to describe an MPC, represent the projection of a point in the camera coordinate system on a microlens using 6 parameters and the knowledge of the microlens center coordinates on the raw image by

$$\begin{bmatrix}\Delta p\\ \Delta g\end{bmatrix} = \frac{1}{K_1\, z + K_2}\begin{bmatrix} f_x\, x - z\, \dot{p}_c \\ f_y\, y - z\, \dot{g}_c\end{bmatrix} \quad , \tag{5.19}$$

where K_1 and K_2 are additional intrinsic parameters to the conventional pinhole camera model [48], $\Delta p = p - p_c$ and $\Delta g = g - g_c$ with (p_c, g_c) defining the microlens center coordinates associated with the raw image coordinates (p, g), and $\dot{p}_c = p_c - c_x$ and $\dot{g}_c = g_c - c_y$. The $\left(f_x, f_y, c_x, c_y\right)$ are the parameters used to convert normalized coordinates to image coordinates. This model can be rewritten to get a pinhole-like representation by isolating the coordinates of the point $[x, y, z]^T$. This allows to define the intrinsic matrix $\mathbf{K}_b$ and the translation vector $\mathbf{t}_b$ for the microlens camera (p_c, g_c) as

$$\mathbf{K}_b = \begin{bmatrix} \frac{f_x}{K_1} & 0 & -\frac{\dot{p}_c}{K_1} \\ 0 & \frac{f_y}{K_1} & -\frac{\dot{g}_c}{K_1} \\ 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{t}_b = \begin{bmatrix} \frac{K_2\, \dot{p}_c}{K_1\, f_x} \\ \frac{K_2\, \dot{g}_c}{K_1\, f_y} \\ \frac{K_2}{K_1} \end{bmatrix} \quad . \tag{5.20}$$

The translation vector $\mathbf{t}_b$ allows to define the position of the microlens camera relatively to the camera coordinate system origin which corresponds to the plane containing the viewpoint projection centers [21]. For considering the relationship with the world coordinate system, one should consider the projection matrix defined as

$$\mathbf{P}_b = \mathbf{K}_b \begin{bmatrix}\mathbf{I}_{3\times 3} & \mathbf{t}_b\end{bmatrix} {}^{c}\mathbf{T}_w \tag{5.21}$$

where ${}^{c}\mathbf{T}_w$ defines the rigid body transformation between the world and camera coordinate systems. Representing the raw image coordinates by the 4D coordinates of the rays in the image space using $i = \Delta p$, $j = \Delta g$, and (k, l) using the rectangular sampling basis proposed in Section 3.5, such that $[p_c, g_c]^T = \operatorname{diag}\left(\frac{d_h}{2}, d_v\right)[k, l]^T + [p_0, g_0]^T$ (Figure 5.4.b), the intrinsic and translation vector can be redefined as

$$\mathbf{K}_b = \begin{bmatrix} \frac{f_x}{K_1} & 0 & -\frac{\dot{c}_x}{K_1} - k\,\frac{1}{2}\frac{d_h}{K_1} \\ 0 & \frac{f_y}{K_1} & -\frac{\dot{c}_Y}{K_1} - l\,\frac{d_v}{K_1} \\ 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{t}_b = \begin{bmatrix} \frac{K_2}{K_1}\left(\frac{\dot{c}_x}{f_x} + k\,\frac{1}{2}\frac{d_h}{f_x}\right) \\ \frac{K_2}{K_1}\left(\frac{\dot{c}_y}{f_y} + l\,\frac{d_v}{f_y}\right) \\ \frac{K_2}{K_1} \end{bmatrix} \quad . \tag{5.22}$$

In this mapping, d_h and d_v correspond to the horizontal and vertical distances between consecutive microlenses centers, (p_0, g_o) correspond to the origin for the (k, l) coordinates in the raw image, and $\dot{c}_x = p_0 - c_x$ and $\dot{c}_y = g_0 - c_y$.

Similarly to the microlens intrinsic matrix (5.8), one can obtain the mapping to the representation in (5.2). For the principal point, one has $[u_0, v_0]^T = -\left[\dot{c}_x/K_1, \dot{c}_y/K_1\right]^T$ and $[\Delta u_0, \Delta v_0]^T = -\left[d_h/2K_1, d_v/K_1\right]^T$. The scale factors are defined as $k_u = f_x/K_1$ and $k_v = f_y/K_1$. Finally, the baseline is $[\Delta x_0, \Delta y_0, 0]^T = K_2/K_1\ [d_h/2f_x, d_v/f_y, 0]^T$ and the location of the microlens camera array is $[x_0, y_0, z_0]^T = K_2/K_1$ $\left[\dot{c}_x/f_x, \dot{c}_y/f_y, 1\right]^T$. Looking at these definitions, one can identifiy the same relationships as in (5.14) with $\left[b_x, b_y, 0\right]^T = \left[-K_2/f_x, -K_2/f_y, 0\right]^T$.

The LFIM associated with the camera model of Bok *et al.* [21], considering the plane containing the viewpoint projection centers as the origin of the camera coordinate system, is defined as

$$\mathbf{H}_b = \begin{bmatrix} -\frac{K_2}{f_x} & 0 & 0 & 0 & 0 \\ 0 & -\frac{K_2}{f_y} & 0 & 0 & 0 \\ \frac{K_1}{f_x} & 0 & \frac{1}{2}\frac{d_h}{f_x} & 0 & \frac{\dot{c}_x}{f_x} \\ 0 & \frac{K_1}{f_y} & 0 & \frac{d_v}{f_y} & \frac{\dot{c}_y}{f_y} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad . \tag{5.23}$$

The microlens projection matrix (5.20) allows to identify an incorrect definition for the extrinsic parameters when the z-component of the translation is negative in the calibration procedure proposed by Bok *et* *al.* [21]. Namely, in this situation, one should change the signs of $\mathbf{r}_1$, $\mathbf{r}_2$, ${}^{c}\mathbf{t}_w$ and K_2. K_1 is related with the scale factors k_u and k_v in (5.1) and therefore its sign should not be changed.

5.5 Corner-based Calibration

The methods to calibrate MPCs are scarce and normally consider features in MIs. The calibration methods differ mainly in the camera model (Table 5.1). More specifically, Nousias *et al.* [115] considered the independent calibration of each microlens type described by the camera model proposed by Bok *et al.* [21] for SPCs. The independent calibration requires the estimation of a high number of parameters being the majority of them redundant. Nousias *et al.* [115] acknowledged the existence of common extrinsics among the microlens types but did not proposed a simultaneous calibration of the different microlens types.

In the camera model proposed (Section 5.5.1), one considers common intrinsic and extrinsic parameters among the microlens types. The difference among the microlens types is accounted by the blur model that describes the different defocus behaviors according to the microlens world focal planes. This allows to represent an MPC with a reduced number of parameters and perform a simultaneous calibration of the microlens types.

Method	Number LFIM Parameters	Number Extrinsic Parameters	Blur Model	Features
Nousias *et al.* [115]	$6M$	$6PM$	No	Corners in MIs
Monteiro *et al.* [104]	8	$6P$	Yes	Corners and blur radius in MIs

Table 5.1: State of the art comparison for MPC calibration procedures. P denotes the number of poses and M denotes the number of microlens types.

The accuracy of the camera calibration also relies on the precision of the detected features in the raw images. The corner detection in the MIs of an MPC is particularly challenging given the different microlens types and defocus blur.

Different solutions addressing this problem have been explored, usually exploiting special physical targets or techniques to recreate favourable conditions for the feature detection. Heinze *et al.* [66] considered a special calibration target with circular patterns to help avoiding incorrect matches along epipolar lines in the depth estimation process. Bok *et al.* [21] claimed that due to the MI small size, corners cannot be accurately detected, and, therefore, edge features of a checkerboard pattern are detected and used for calibration. This approach cannot handle different microlens types [115]. Nousias *et al.* [115] operates corners detection on MIs and is able to categorize different microlens types. Corners are found at the saddle point between the two regions of maximum and minimum intensity, ensuring more robustness against blurred MIs. Although outperforming classical state of the art like Harris [62] or FAST [125] corner detectors, it leaves margin for improvement.

Thus, is proposed a detector that separately estimates corner location and radius blur in pixels (Section 5.5.2). The corner location estimation is based on intensity analysis of the boundaries of a window centered around each corner to ensure robustness against different degrees of defocus, while the blur calculation makes use of a conventional focus measure from the literature [119] which is adapted and calibrated for giving a metric in pixels.

5.5.1 MPC Camera Model

In the previous sections, was described the camera model of the microlens camera array composed of identical microlenses in an FPC, however, an MPC has several types of microlenses each with a different focal length. Strobl *et al.* [132] highlighted the need to model the different microlens types to accurately represent an MPC. Nonetheless, the only known works that model the different microlens types are [66, 115]. Considering the thin lens equation to describe a microlens and a fixed distance d^I_μ, one can see that each microlens type has a different focal plane. Alternatively, for having a point in the main lens focal plane in focus, each microlens type needs to have a different spacing between the image sensor and the microlens array (d_μ^I, d_μ^{II}, and d_μ^{III} in Figure 5.1.b). However, there is only one image sensor so some microlenses produce blurred images of the point. Additionally,the features extracted from the MIs refer to the actual single image sensor at distance d_μ^I in the MPC and not the virtual image sensors d_μ^{II} and d_μ^{III}.

The camera models used for plenoptic cameras consider the microlenses as pinholes [21, 41, 104, 150]. The pinhole model accurately represents the chief-ray originating at a given 3D point. This chief-ray does not depend on the microlens focal plane and detecting its position in the blurred MIs poses a challenge. Thus, in this section, is proposed a camera model that describes the point projections of a world point in the different microlenses using a single LFIM (4.17) and the specific defocus behavior of each microlens type using the blur radius b derived from the models [11, 18]

$$b = k_s \left| \frac{1}{\mathbf{t}_3 \tilde{\mathbf{m}}} - \frac{1}{z_\Omega} \right| \tag{5.24}$$

with $k_s = w\ d/2$ where w is the distance between the microlens and the image sensor, d is the microlens aperture and z_Ω is the depth of the microlens focal plane in the camera coordinate system. $\mathbf{t}_3\tilde{\mathbf{m}}$ is the depth of the point $\mathbf{m} = [x, y, z]^T$ in the camera coordinate system where $\mathbf{t}_3$ corresponds to the third row of ${}^c\mathbf{T}_w$. This allows to represent an MPC using an affine mapping with 7 parameters (not considering the restriction (5.18)) and a blur model with 1 common scale parameter and 1 additional parameter for each microlens type (depth of the microlens focal plane). The camera model proposed allows to define a calibration procedure for an MPC using corner points and their corresponding blur in each microlens as features.

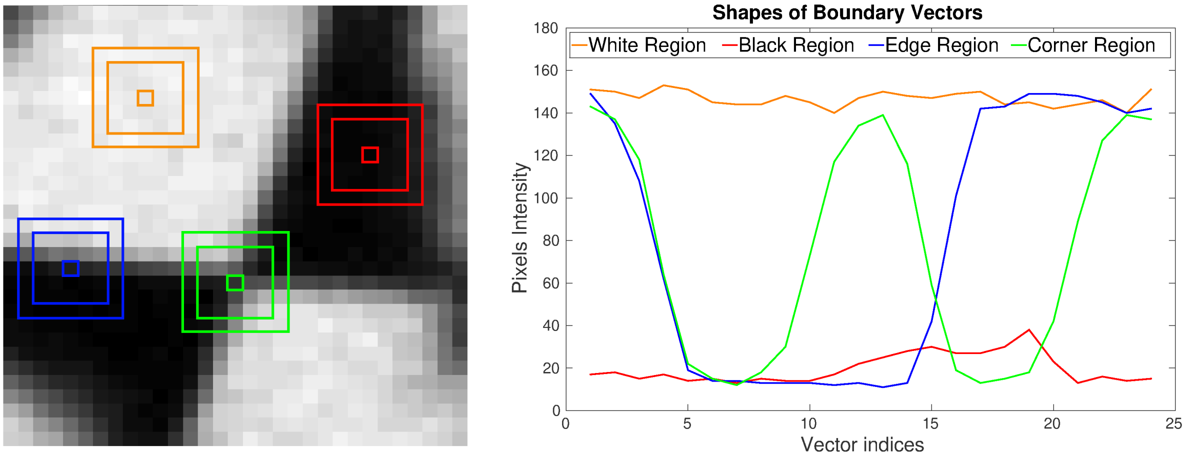
5.5.2 Corner and Blur Radius Detection

In this section, is provided a method capable of detecting and clustering the imaged corner points from conventional checkerboards. The detection is applied directly on the MIs, avoiding dependencies from pre-processing or depth information. The proposed algorithm has three steps: (i) obtain a likelihood map of the corners location using a boundary centered around a pixel candidate similar to the one used in [20], then (ii) create clusters of the imaged points belonging to the same checkerboard corner, and finally (iii) fit lines to estimate the exact position of the imaged corners within each single cluster. Moreover, one emphasizes the importance of accounting for the different microlens types and the different degrees of defocus in each MI. The proposed solution is to model this separately from the corner points' location by means of a blur radius detector based in the Tenengrad Variance focus measure [119].

Likelihood Map Step. The first step of the corner detection consists in the generation of a likelihood map where each pixel value indicates the probability of that pixel containing a corner. Such map can be used to extract the actual corners or as an initial estimation for a refinement process.

The corner points are searched in the MIs. This requires the knowledge about the microlens centers position but allows to avoid dark areas between MIs. This is solved using a white image and the methods described in [21, 41] or using the procedure described in [66], common to Raytrix RxLive software [123] as in [117]. The search operation in the MIs has a large computational effort due to the large number of MIs in the raw images. Therefore, a pre-processing step is performed to obtain the probability of the presence of a corner in each MI by looking at the ratio of dark and white pixels and the presence of lines at different angles.

The calculation of the pixelwise likelihood map is performed only on the MIs classified as possible candidates. The proposed method takes inspiration from previous works, namely, Bok *et al.* [20] analyzed the circular boundaries assuming a sharp change between black and white regions that do not happen for blurred images, and Nousias *et al.* [115] used lines towards the highest intensity points to overcome this issue. The likelihood score is computed merging the two ideas, namely selecting the boundaries of a squared window around each point and analyzing the curve of the intensity of its values. Ideally, a linear vector obtained from the boundary should exhibit a particular shape consisting of two distinct maxima and minima values approximately at the same distance between each other, being half of the vector length, as visible in Figure 5.2.



(a) MI

(b) Boundary profiles of points in (a)

Figure 5.2: Example of a MI with a corner. Four different regions **(a)** are analyzed to exhibit their characteristic shapes **(b)**: white and black textureless areas, an edge and a corner region.

The likelihood score is calculated using two penalty functions that reduce the score when the vector shape differs from the ideal one:

$$l(p) = 1 - \rho_I - \rho_p \tag{5.25}$$

with

$$\rho_I = \sum_{k=1,2} \frac{|I_{m,k} - I_m|}{\sigma_{I,m}} + \frac{|I_{M,k} - I_M|}{\sigma_{I,M}} \quad \text{and} \tag{5.26}$$

$$\rho_p = \frac{|(p_{m,2} - p_{m,1}) - \lfloor \frac{l_v}{2} \rfloor|}{\sigma_{d,m}} + \frac{|(p_{M,2} - p_{M,1}) - \lfloor \frac{l_v}{2} \rfloor|}{\sigma_{d,M}} \tag{5.27}$$

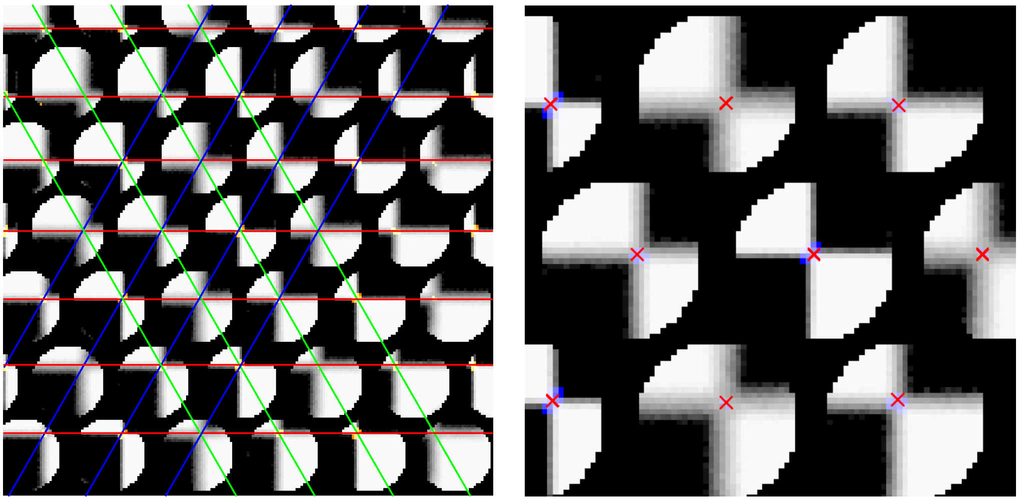
where $l(p)$ denotes the likelihood score of pixel p in the linear vector and (ρ_I, ρ_p) are the two penalty functions calculated using the difference between the intensity and relative position values of the minima and maxima and the ideal ones. The m indicates the minima and M the maxima points, I the intensity and p the position in the linear vector. $\sigma_{d,M}$, $\sigma_{d,m}$, $\sigma_{I,M}$, $\sigma_{I,m}$ are fixed value variables to control the contribution of each penalty function. I_m and I_M are respectively the minimum and maximum intensity values of the whole image, and $\frac{l_v}{2}$ is half of the vector length.

Finally, in order to avoid assigning scores to pixels that do not actually represent a corner, one performs a connected component analysis on a binary version of the likelihood map where all pixels with likelihood greater than zero are selected. If there are more than one unconnected components in the same MI, one evaluates their score as the sum of the likelihood of their points and keep the highest one.

Clustering Step. In this step, one requires the 2D coordinates of the imaged corners. Hence, one transforms the likelihood map into points by selecting for each connected component a weighted average position of the corresponding pixels using the likelihood scores as weights. Before the actual clustering, one filters the points by means of a statistical outliers removal. Outliers are defined as points that do not have enough neighbors within a predefined range. At the same time, one builds a grid with a rough guess of where the clusters centers are to facilitate the convergence of the clustering algorithm. This step is not actually required, yet it significantly reduces the probability of incurring into wrong clustering and the number of iterations needed to reach the final solution.

Following these two steps, one is able to provide as input for the clustering an outlier-free ensemble of points and a rough initial guess of the grid centers. The *k*-means algorithm has been chosen for the final clustering using the euclidean distance as measure for the clusters classification.

Line Constraints Refinement. The final step is a region-based process, repeated for every cluster. The operation explained in this section is performed on points within the same cluster. At this step, one does not need 2D coordinates so the likelihood map is again used to achieve higher accuracy in the selection of the final points.



(a) Lines drawn on the MIs (b) Detected corners

Figure 5.3: Example of lines within a single cluster and detected corners. In **(a)**, each epipolar line is shown in a different color. In **(b)**, the likelihood map is shown in a scale of blues and the corners detected after the refinement with red crosses.

The epipolar geometry states that the corresponding corners must lie on three epipolar lines with 0, 60 and 120 $\deg$ inclination since the microlenses are arranged on an hexagonal grid, assuming rectified images. Lines can be defined by slope and y-intercept. By fixing the slope and tuning the y-intercept value, a score is calculated accumulating the likelihood of the points that lie on each generated line, *e.g.* a higher score indicates that a line is crossing more high probability points. Intuitively, the correct lines should be those lines that cross the pixels with higher probability. In order to choose the correct lines and avoid false positives, one must ensure a minimum distance between them to prevent adjacent lines to be chosen together. This distance is set to be the radius of a MI.

The corners lie in the intersection of the generated lines. In an ideal situation, the three lines would intersect in the same point, however, in the real case, the lines intersect at different points. A corner is detected if the distance between the different intersection points is smaller than a predefined tolerance threshold. Here, there is a tradeoff, a larger tolerance allows to select more points at the price of reducing the accuracy while a narrow tolerance increases the accuracy but reduces the number of selected points.

Blur Calibration and Estimation. In this step, one estimates the blur radius in pixels for each detected corner. The literature shows different approaches for the measurement of blur but normally there is no connection to a precise estimation of the blur radius in pixels. For example, in [115], the focus measure was only used to classify the microlens type.

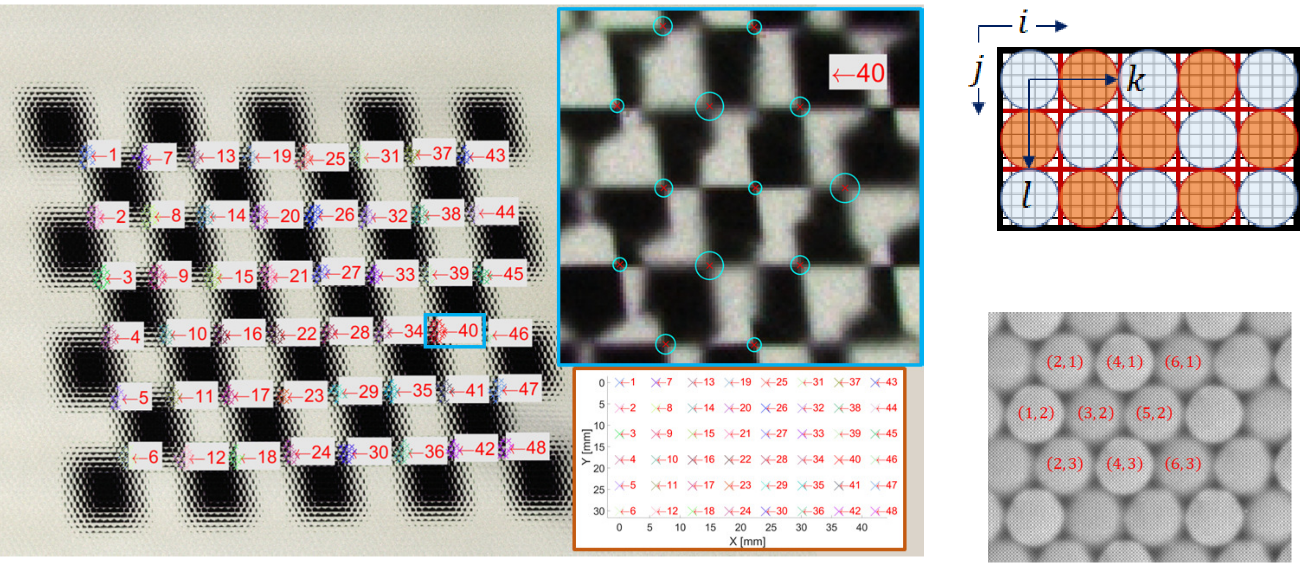
The approach consists in first estimating a mapping function from a focus measure, the Tenengrad Variance [119], to the blur radius defined in pixels. Taking inspiration from a similar idea [88], one generates an *ad-hoc* dataset of synthetically generated MIs with a corner where one gradually adds an increasing amount of blur to simulate all possible blur patterns to reliably map the blur measurements in pixels. A total of 21 images are used considering a blur radius ranging from 0 to 10 pixels with a step of 0.5 pixels. To emulate real images one also adds Gaussian noise with zero mean and variance $\sigma_n^2 = 0.0003$.

The focus measure is obtained using a small region around the detected corner instead of using the whole MI. The remaining part of the image should not affect the estimation since corners at the edge of a microlens may have less texture and thus obtain different focus measurements.

After estimating the mapping function, one uses this function to map the focus measure associated with a corner to the corresponding blur radius in pixels.

5.5.3 Calibration Procedure

The calibration proposed considers the corners of a planar calibration grid of known dimensions and the corresponding blur radius as features (Figure 5.4.a). In the following, is assumed that the microlenses centers and types are known [41, 115] and that the corners in the world coordinate system have been matched with the imaged corners. An imaged corner is defined by a ray $\mathbf{\Phi} = [i, j, k, l]^T$ in the image space. The (i, j) coordinates correspond to the pixel coordinates of the detected corners on the MIs relatively to the corresponding microlens center. The (k, l) coordinates correspond to the microlens coordinates considering a rectangular sampling to represent the microlens center coordinates in the raw image (Figure 5.4.b).



(a) (b)

Figure 5.4: Features of dataset acquired with a Raytrix camera. The corners and blur radius detected on the raw image and the clustering performed is highlighted in **(a)**. A detail of cluster 40 is shown in the blue rectangle. The association of the clusters with the 3D points is depicted in the orange rectangle. The rectangular coordinate system to represent the microlens centers in the raw image is depicted in **(b)**.

5.5.4 Linear Initialization

In this section, is considered the mapping in Section 5.3 to define a linear solution for the microlens array parameters associated with a plenoptic camera and the extrinsic parameters for each pose of the calibration grid. The blur model (5.24) associated with each microlens type is used to define the microlens focal planes. The linear solution comprises homography, intrinsic, extrinsic and blur parameters estimation steps.

Homography Estimation. Considering the microlens projection matrix $\mathbf{P}^{kl}$ (5.1) with $\mathbf{K}^{kl}$ (5.8) and $\mathbf{t}^{kl}$ (5.13), a point $\mathbf{m} = [x, y, z]^T$ in the object space is projected to a point in the image plane $\mathbf{q} = [i, j]^T$ by

$$\tilde{\mathbf{q}} \sim \mathbf{P}^{kl}\,\tilde{\mathbf{m}} = \mathbf{K}^{kl}\left[{}^{c}\mathbf{R}_{w} \;\; {}^{c}\mathbf{t}_{w} + \mathbf{t}^{kl}\right]\tilde{\mathbf{m}} \tag{5.28}$$

where the symbol $\sim$ denotes equal up to a scale factor. The coplanar grid points allow to define a world coordinate system such that the z-coordinate is zero. In this context, denoting $\tilde{\mathbf{m}} = [x, y, 1]^T$, one can redefine the projection (5.28) as $\tilde{\mathbf{q}} \sim \mathbf{H}^{kl}\,\tilde{\mathbf{m}}$ where

$$\mathbf{H}^{kl} = \mathbf{K}^{kl}\left[\mathbf{r}_1 \;\; \mathbf{r}_2 \;\; {}^{c}\mathbf{t}_{w} + \mathbf{t}^{kl}\right] \tag{5.29}$$

is the parametric homography matrix for the microlens camera (k, l), and ${}^{c}\mathbf{R}_{w} = [\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3]$.

The homography matrix $\mathbf{H}^{kl}$ like the projection matrix $\mathbf{P}^{kl}$ changes among microlenses as a result of the principal point shift and baseline defined in Section 5.1. Let us consider that $\mathbf{H}^{kl}$ can be defined from the homography matrix $\mathbf{H}^{0}$ associated with the microlens coordinates $(k, l) = (0, 0)$ and the homography microlens change matrix $\mathbf{A}^{kl}$ by

$$\mathbf{H}^{kl} = \underbrace{\begin{bmatrix} h^0_{11} & h^0_{12} & h^0_{13} \\ h^0_{21} & h^0_{22} & h^0_{23} \\ h^0_{31} & h^0_{32} & h^0_{33} \end{bmatrix}}_{\mathbf{H}^0} + \begin{bmatrix} k & 0 & 0 \\ 0 & l & 0 \\ 0 & 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 0 \end{bmatrix}}_{\mathbf{A}^{kl}}. \tag{5.30}$$

Considering the homography projection of a calibration grid corner $\tilde{\mathbf{m}} =$

$[x, y, 1]^T$ in the object space to the image point $\tilde{\mathbf{q}}$ for the microlens camera (k, l), applying the cross product by $\tilde{\mathbf{q}}$ on each side of the projection equation leads to $[\tilde{\mathbf{q}}]_\times \mathbf{H}^{kl}\, \tilde{\mathbf{m}} = \mathbf{0}_{3\times 1}$, where $\left[(\cdot)\right]_\times$ is a skew-symmetric matrix that applies the cross product. Using the properties of the Kronecker product [93] and solving for each of the unknown parameters, one obtains

$$\left(\tilde{\mathbf{m}}^T \otimes [\tilde{\mathbf{q}}]_\times\right) \mathbf{T} \begin{bmatrix} \mathbf{h}^0 \\ \mathbf{a}^{kl} \end{bmatrix} = \mathbf{0}_{3\times 1} \tag{5.31}$$

where

$$\mathbf{T} = \left[\begin{array}{c cccccc} & k & 0 & 0 & 0 & 0 & 0 \\ & 0 & l & 0 & 0 & 0 & 0 \\ & \multicolumn{6}{c}{\mathbf{0}_{1\times 6}} \\ & 0 & 0 & k & 0 & 0 & 0 \\ \mathbf{I}_{9\times 9} & 0 & 0 & 0 & l & 0 & 0 \\ & \multicolumn{6}{c}{\mathbf{0}_{1\times 6}} \\ & 0 & 0 & 0 & 0 & k & 0 \\ & 0 & 0 & 0 & 0 & 0 & l \\ & \multicolumn{6}{c}{\mathbf{0}_{1\times 6}} \end{array}\right] , \tag{5.32}$$

and $\mathbf{h}^0$ and $\mathbf{a}^{kl}$ correspond to vectorizations of the matrix $\mathbf{H}^0$ and $\mathbf{A}^{kl}$ by stacking their columns and removing the zero entries, respectively. The solution $\left[\mathbf{h}^0, \mathbf{a}^{kl}\right]^T$ for the parametric homography matrix can be estimated using Singular Value Decomposition (SVD).

The parametric homography matrix (5.30) is defined using 15 parameters. According to (5.31), each point correspondence $(\tilde{\mathbf{m}}, \tilde{\mathbf{q}})$ originates three equations with only two being linearly independent. Nonetheless, the restrictions on the microlens camera array also originate restrictions on the projections of a point in the object space. Namely, the ray in the image space $\mathbf{\Phi}^{kl} = [i, j, k, l]^T$ associated with an arbitrary microlens (k, l) can be described from the ray coordinates $\mathbf{\Phi}^0 = [i_0, j_0, 0, 0]^T$ associated with the microlens $(k, l) = (0, 0)$ by $\mathbf{\Phi}^{kl} = \mathbf{\Phi}^{0} + [k\beta, l\beta, k, l]^{T}$, where β corresponds to the disparity of the point defined on the MIs. This reduces the number of linearly independent equations originated by a point in the object space to 4. Thus, one needs at least 4 non-collinear points to obtain the entries of the homography matrix $\mathbf{H}^{kl}$. In the estimation of the homography matrix $\mathbf{H}^{kl}$ one should also consider the practical aspects mentioned in Section 4.3.1.

Intrinsic and Extrinsic Estimation. The structure of the homography matrix (5.29) in conjunction with the orthogonality and identity of the column vectors of ${}^{c}\mathbf{R}_{w}$ allow to define constraints on the intrinsic parameters as $\mathbf{h}_1{}^{T}\mathbf{B}^{kl}\,\mathbf{h}_2 = 0$ and $\mathbf{h}_1{}^{T}\mathbf{B}^{kl}\,\mathbf{h}_1 - \mathbf{h}_2{}^{T}\mathbf{B}^{kl}\,\mathbf{h}_2 = 0$ [151] where $\mathbf{h}_m$ refers to the m-th column vector of $\mathbf{H}^{kl}$, and the symmetric matrix that describes the image of the absolute conic is defined as $\mathbf{B}^{kl} = \mathbf{K}^{kl^{-T}}\mathbf{K}^{kl^{-1}}$ [92, 151]. Using the knowledge of the intrinsic matrix $\mathbf{K}^{kl}$, one can represent the absolute conic $\mathbf{B}^{kl}$ for a microlens camera (k, l) using a minimal number of parameters.

The intrinsic matrix $\mathbf{K}^{kl}$ differs on the principal point for each microlens leading to different images of the absolute conic. The principal points change regularly between consecutive microlenses by the principal point shift $[\Delta u_0, \Delta v_0]^{T} = \left[-\frac{h_{uk}}{h_{ui}}, -\frac{h_{vl}}{h_{vj}}\right]^{T}$ which can be used to constraint the parametric representation of $\mathbf{B}^{kl}$. Namely, considering (5.8), $\mathbf{B}^{kl}$ can be defined as

$$\mathbf{B}^{kl} = \mathbf{B}^{0} + k\,\mathbf{C}^{k} + l\,\mathbf{D}^{l} + k^{2}\,\mathbf{E}^{k} + l^{2}\,\mathbf{F}^{l} \tag{5.33}$$

with

$$\mathbf{B}^{0} = \begin{bmatrix} h_{ui}^{2} & 0 & h_u h_{ui} \\ 0 & h_{vj}^{2} & h_v h_{vj} \\ h_u h_{ui} & h_v h_{vj} & 1 + h_u^{2} + h_v^{2} \end{bmatrix}, \tag{5.34}$$

$$\mathbf{C}^k = \begin{bmatrix} 0 & 0 & h_{ui}h_{uk} \\ 0 & 0 & 0 \\ h_{ui}h_{uk} & 0 & 2h_u h_{uk} \end{bmatrix}, \ \mathbf{E}^k = \begin{bmatrix} & \mathbf{0}_{2\times 3} & \\ 0 & 0 & h_{uk}^2 \end{bmatrix}, \tag{5.35}$$

$$\mathbf{D}^l = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & h_{vj}h_{vl} \\ 0 & h_{vj}h_{vl} & 2h_v h_{vl} \end{bmatrix}, \text{ and } \mathbf{F}^l = \begin{bmatrix} & \mathbf{0}_{2\times 3} & \\ 0 & 0 & h_{vl}^2 \end{bmatrix}. \tag{5.36}$$

This allows to define a representation for $\mathbf{B}^{kl}$ using 11 distinct non-zero entries $\mathbf{b}^{kl} = [\, b_{11}, b_{13}, b_{22}, b_{23}, b_{33}, c_{13}, c_{33}, d_{23}, d_{33}, e_{33}, f_{33}\,]^T$ where $(\cdot)_{nm}$ represents the entry in row n and column m of the matrix $(\cdot)$. Considering these parameters, the intrinsic parameters constraints can be redefined as

$$\begin{bmatrix} h_{11}h_{12} & {h_{11}}^2 - {h_{12}}^2 \\ h_{11}h_{32} + h_{12}h_{31} & 2\left(h_{11}h_{31} - h_{12}h_{32}\right) \\ h_{21}h_{22} & {h_{21}}^2 - {h_{22}}^2 \\ h_{21}h_{32} + h_{22}h_{31} & 2\left(h_{21}h_{31} - h_{22}h_{32}\right) \\ h_{31}h_{32} & {h_{31}}^2 - {h_{32}}^2 \\ k\left(h_{11}h_{32} + h_{12}h_{31}\right) & 2k\left(h_{11}h_{31} - h_{12}h_{32}\right) \\ k\left(h_{31}h_{32}\right) & k\left({h_{31}}^2 - {h_{32}}^2\right) \\ l\left(h_{21}h_{32} + h_{22}h_{31}\right) & 2l\left(h_{21}h_{31} - h_{22}h_{32}\right) \\ l\left(h_{31}h_{32}\right) & l\left({h_{31}}^2 - {h_{32}}^2\right) \\ k^2\left(h_{31}h_{32}\right) & k^2\left({h_{31}}^2 - {h_{32}}^2\right) \\ l^2\left(h_{31}h_{32}\right) & l^2\left({h_{31}}^2 - {h_{32}}^2\right) \end{bmatrix}^T \mathbf{b}^{kl} = \mathbf{0}_{2\times 1}\,. \tag{5.37}$$

Normally, each homography generates 2 equations for determining the matrix of the absolute conic image [151]. The parametric representation (5.30), representing an arbitrary microlens (k, l), generates 6 equations. Nonetheless, only 2 equations are independent regarding the entries of

$\mathbf{B}^0$, so one needs to acquire at least 3 calibration grid poses to estimate $\mathbf{b}^{kl}$ defined up to a scale factor.

The intrinsic matrix parameters can be recovered from $\mathbf{B}^{kl}$. More specifically, rewriting the intrinsic matrix $\mathbf{K}^{kl}$ (5.8) as

$$\mathbf{K}^{kl} = \underbrace{\begin{bmatrix} k_u & 0 & u_0 \\ 0 & k_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{K}^0} + \begin{bmatrix} k & 0 & 0 \\ 0 & l & 0 \\ 0 & 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} & \Delta u_0 \\ \mathbf{0}_{3\times 2} & \Delta v_0 \\ & 0 \end{bmatrix}}_{\mathbf{G}^{kl}}, \tag{5.38}$$

one can define $\mathbf{B}^0 = \mathbf{K}^{0^{-T}}\mathbf{K}^{0^{-1}}$. This allows to estimate the entries of $\mathbf{K}^0$ using the Cholesky decomposition of $\mathbf{B}^0$ and correcting the scale factor considering $k^0_{33} = 1$. The principal point shift can be estimated considering $\Delta u_0 = -\frac{h_{uk}}{h_{ui}} = -\frac{c_{13}}{b_{11}}$ and $\Delta v_0 = -\frac{h_{vl}}{h_{vj}} = -\frac{d_{23}}{b_{22}}$.

The extrinsic parameters can be estimated once the intrinsic matrix $\mathbf{K}^{kl}$ is known. From (5.29), the rotation matrix ${}^c\mathbf{R}_w = [\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3]$ is recovered considering

$$\mathbf{r}_1 = \lambda \mathbf{K}^{kl^{-1}}\mathbf{h}_1 \text{ , } \mathbf{r}_2 = \lambda \mathbf{K}^{kl^{-1}}\mathbf{h}_2 \text{ , and } \mathbf{r}_3 = \mathbf{r}_1 \times \mathbf{r}_2 \tag{5.39}$$

with $\lambda = 1/\left\|\mathbf{K}^{kl^{-1}}\mathbf{h}_1\right\| = 1/\left\|\mathbf{K}^{kl^{-1}}\mathbf{h}_2\right\|$. The translation ${}^c\mathbf{t}_w$ and projection center $\mathbf{t}^{kl}$, considering the microlens pinhole constraint (5.6), are recovered solving the following system of equations

$$\lambda \mathbf{h}_3 = \begin{bmatrix} \mathbf{K}^{kl} & \mathbf{K}^{kl}\mathbf{J} \end{bmatrix} \begin{bmatrix} {}^c\mathbf{t}_w \\ b_x \end{bmatrix} \tag{5.40}$$

with

$$\mathbf{J} = \begin{bmatrix} u_0 + k\Delta u_0 \\ (v_0 + l\Delta v_0)\,\frac{k_u}{k_v} \\ -k_u \end{bmatrix} \quad . \tag{5.41}$$

Blur Estimation. The blur model defines the defocus that occurs in each microlens depending on the distance of the point to the microlens focal plane. The depth of the microlens focal plane corresponds to the depth of the points with blur radius equal to zero (5.24), *i.e.* $z_\Omega = \mathbf{t}_3\tilde{\mathbf{m}}$ for $b = 0$. Normally, the blur radius is not zero due to pixel discretization so one should consider a range for selecting the points with zero blur radius and take the median of the points depth to estimate the microlens focal plane. Once the microlens focal plane depth is known, the parameter k_s is estimated simply taking the median of $k_s = \frac{1}{b}\left|\frac{1}{\mathbf{t}_3\tilde{\mathbf{m}}} - \frac{1}{z_\Omega}\right|$.

5.5.5 Nonlinear Optimization

In this section, the linear solution is refined and radial distortion [25] is considered on the coordinates (u, v). Namely, the undistorted rays in the object space $\mathbf{\Psi}^u = [s, t, u^u, v^u]^T$ are defined from distorted rays in the object space $\mathbf{\Psi} = [s, t, u, v]^T$ by (4.34) that is described by the distortion vector $\mathbf{d} = (k_1, k_2, k_3, b_u, b_v)$. In the nonlinear optimization, one minimizes the reprojection error $\Theta(\cdot)$ and the blur radius error $\tau(\cdot)$ simultaneously for all microlens types

$$\underset{\mathbf{K}^{kl},\mathbf{t}^{kl},\mathbf{b}_m,\mathbf{R}_p,\mathbf{t}_p,\mathbf{d}}{\arg\min} \; \Theta\left(\mathbf{K}^{kl}, \mathbf{t}^{kl}, \mathbf{d}, \mathbf{R}_p, \mathbf{t}_p\right) + \tau\left(\mathbf{b}_m, \mathbf{R}_p, \mathbf{t}_p\right) \; . \tag{5.42}$$

This optimization refines the intrinsic parameters $\mathbf{K}^{kl}$ and $\mathbf{t}^{kl}$, the blur parameters $\mathbf{b}_m = \left[k_s, z_{\Omega_m}\right]^T$, $m = 1, \ldots, M$ where M is the number of microlens types, the extrinsic parameters $\mathbf{R}_p$ (parameterized by Rodrigues formula [48]) and $\mathbf{t}_p$, $p = 1, \ldots, P$ where P is the number of poses, and the distortion vector $\mathbf{d}$.

The reprojection error [63]

$$\Theta\left(\mathbf{K}^{kl}, \mathbf{t}^{kl}, \mathbf{d}, \mathbf{R}_p, \mathbf{t}_p\right) = \sum_{p=1}^{P} \sum_{n=1}^{N_p} \sum_{(k,l)\in\chi_n} \left\| \hat{\mathbf{q}}_n^{kl} - \mathbf{q}_n^{kl} \right\|^2 \tag{5.43}$$

defines the error in pixels using the Euclidean distance between the detected corners $\hat{\mathbf{q}}_n^{kl}$ and the projections $\mathbf{q}_n^{kl}$ of the world coordinate system point $\mathbf{m}_n$ associated with the corner n in the multiple microlens cameras χ_n, *i.e.* $\mathbf{q}_n^{kl} = \Pi\left(\mathbf{K}^{kl}, \mathbf{t}^{kl}, \mathbf{R}_p\, \mathbf{m}_n + \mathbf{t}_p\right)$ where $\Pi(\cdot)$ defines the projection in microlens (k, l) of a point in the camera coordinate system. The detected corners are not directly the ones obtained from the raw image but the projections obtained from the reconstructed point after distortion correction, *i.e.* $\hat{\mathbf{q}}_n^{kl} = \Pi\left(\mathbf{K}^{kl}, \mathbf{t}^{kl}, \eta\left(\mathbf{H}, \mathbf{d}, \mathbf{\Phi}_n\right)\right)$ where η defines the reconstructed point after mapping the ray in the image space $\mathbf{\Phi}_n$ associated with the corner n to the ray in object space (3.1), followed by distortion rectification (4.34) and reconstruction [107]. Notice that $\mathbf{H}$ can be defined from $\mathbf{K}^{kl}$ and $\mathbf{t}^{kl}$ using the mappings defined in Section 5.3. N_p corresponds to the number of corners detected on pose p.

The blur radius

$$\tau\left(\mathbf{b}_m, \mathbf{R}_p, \mathbf{t}_p\right) = \sum_{p=1}^{P} \sum_{m=1}^{M} \sum_{n=1}^{N_p} \sum_{(k,l)\in\chi_n} \left\| \hat{b}_n^{kl} - b_n^{kl} \right\|^2 \tag{5.44}$$

defines the error in pixels using the Euclidean distance between the detected blur radius $\hat{\mathbf{b}}_n^{kl}$ and the blur radius $\mathbf{b}_n^{kl}$ estimated for the point $\mathbf{m}_n$ using (5.24) for the multiple microlens cameras.

The nonlinear optimization is solved using the trust-region-reflective algorithm [35], where a sparsity pattern for the Jacobian matrix is provided. The number of parameters over which one optimizes is 7 for the intrinsic parameters, $M + 1$ for the blur parameters, 5 for the lens distortion parameters, and $6P$ for the extrinsic parameters.

5.5.6 Experimental Results

In this section are presented the results of the corner detector and the calibration procedure proposed. The methodologies proposed are applied to synthetic datasets obtained using the toolbox [101] and to a dataset acquired with a commercially available MPC with three microlens types, the R42 Raytrix with a 50 mm lens.

Synthetic Dataset Results. Synthetic images have successfully proven to emulate plenoptic cameras [101], so one created a dedicated set of synthetic raw images with a checkerboard pattern using the Blender engine.

The knowledge of the three-dimensional position of the pattern and the camera parameters allows to calculate the corners' positions using ray tracing. For every pixel, a bundle of rays is emitted and traced to the scene until they reach the object, fetching its position in the three-dimensional space. Even in the unfocused case, in which the rays may not converge to the same point, their positions can be averaged to robustly recover the coner's positional information [100]. One way to do this is to render a positional image along with the colored image and matching the colored point with its positional information. Rendering the positional image with a higher resolution and picking the closest point, one can reach a sub-pixel accuracy close to 0.1 pixels. This information allows to create a benchmark and evaluate the performance of the corner detection algorithm proposed. For this purpose, a set of 11 images of 3500×3500 pixels with 11017 MIs is created. On average, each image contains 1335.4 corners, for a total of 14689 corners, ensuring the statistical significance of the analysis.

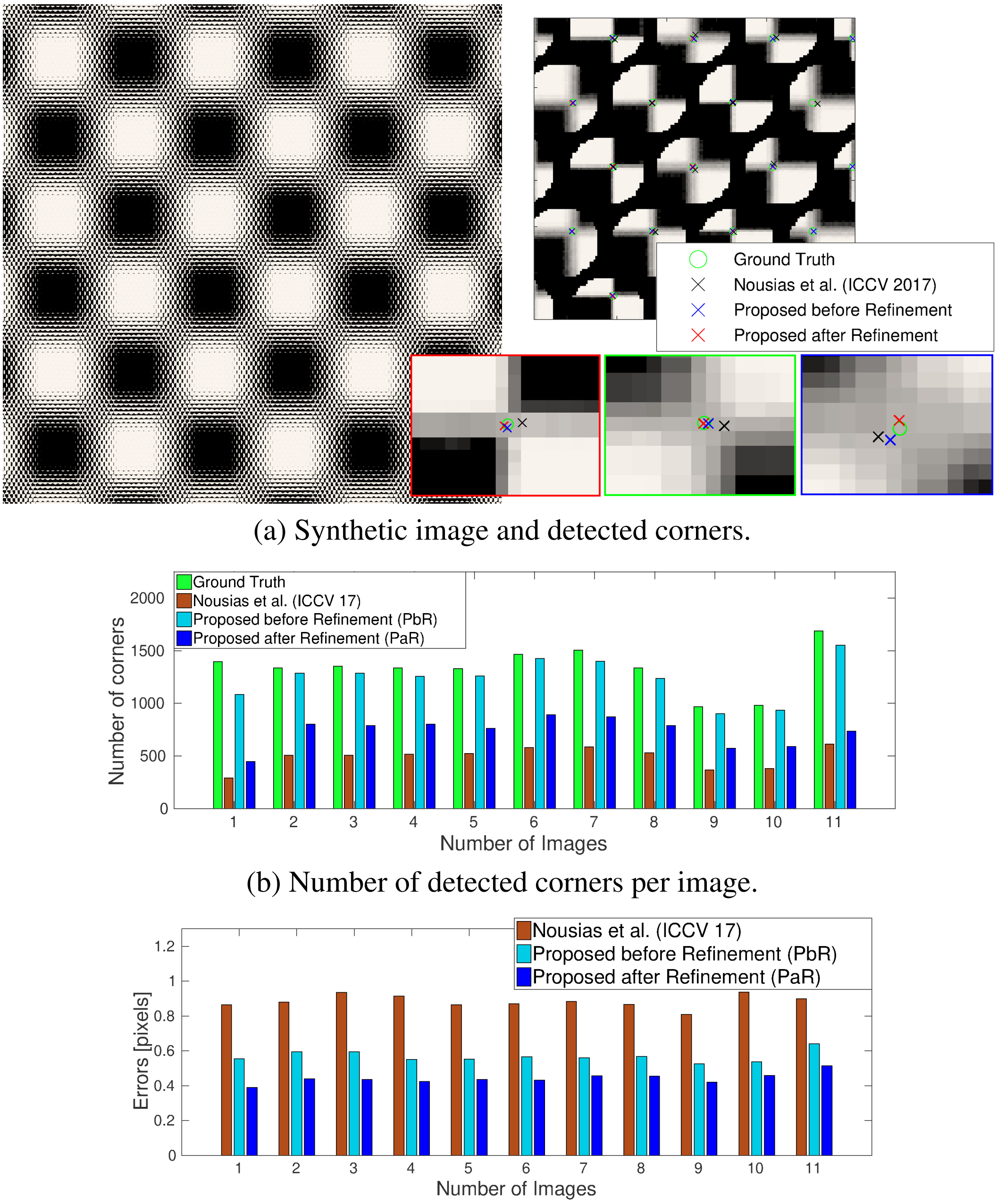
Standard corner detection methods have shown to fail on MIs so one compared the method proposed with the state of the art [115] (denoted as *Nousias17*). Additionally, one shows the proposed method performance before and after the refinement step described in Section 5.5.2 to give a further insight on the corner method proposed. The corner detection error is calculated as the difference in pixels between the estimated corner position and the corresponding ground truth corner. In Figure 5.5.c, the average error for the detected corners in each synthetic image is shown. The method proposed retains a lower error in all the images, and it is possible to see that the refinement step improves the estimation, reducing the detection error in average by 22.05%.

For a meaningful analysis of the error, the number of detected corners has to be taken into account. In Figure 5.5.b, one reports the number of corners detected from each algorithm alongside with the number of ground truth corners for each synthetic image. In Table 5.2, one summarizes the corner detection results indicating the average error between the estimated and ground truth corners and the average number of corners detected per pose. The detection ratio gives the percentage of corners detected with respect to the actual number of corners imaged in the synthetic dataset. The proposed method outperforms the state of the art [115]. As expected, the initial step before refinement aims at detecting all corners, reaching almost the full score. The quantity is then traded with the quality in the refinement step.

Average Results	**Error [pix]**	**Corners Detected**	**Detection Ratio**
Nousias17 [115]	0.8842	491.09	36.88%
Proposed	0.5677	**1237.8**	**92.79%**
Proposed Refined	**0.4420**	731.55	55.29%

Table 5.2: Summary of the average results per pose obtained using the different corner detectors. The highlighted values indicates the best result for each category.

The error of the corner detection can be divided for each microlens type since the MPCs has three different microlens types (Figure 5.6). The corner detection approach used in [115] exhibits a smaller error for the microlens type 0 while the same microlens type seems to be more difficult to precisely estimate using the method proposed before the refinement step. In fact, this is the only situation for which the approach proposed does not perform better. After the refinement, however, the error related with the microlens type 0 is the lowest. Repeating the di-



(a) Synthetic image and detected corners.

(b) Number of detected corners per image.

(c) Ground truth based average corner location error.

Figure 5.5: Synthetic dataset corner detection results. **(a)** shows a checkerboard image with examples of detected corners for different microlens types and different amounts of blur, **(b)** relates to the number of corners found and **(c)** reports the average error in pixels.

vision per microlens type for the number of corners found (Figure 5.7), one observes a lower number of corners for the microlens type 2 while the other microlens types show similar values. These results may be related to the amount of defocus blur and to the particular characteristics of the optics of each microlens.

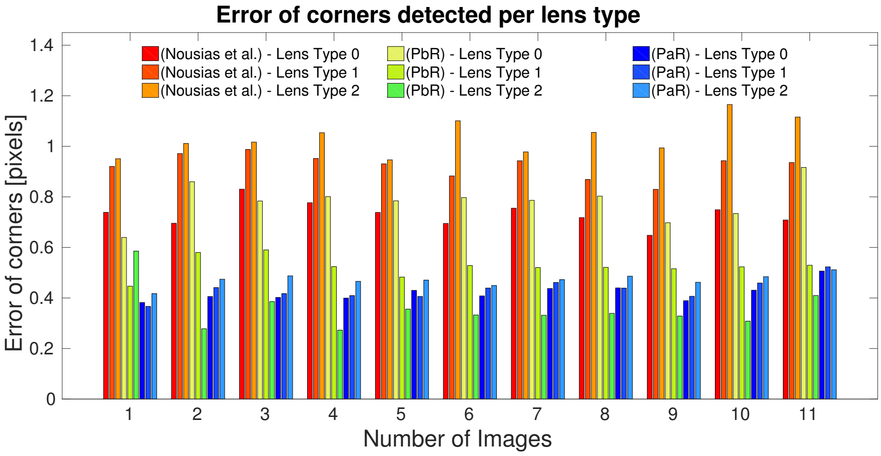


Figure 5.6: Corner detection error for method proposed before (PbR) and after refinement (PaR) step and state of the art method [115].

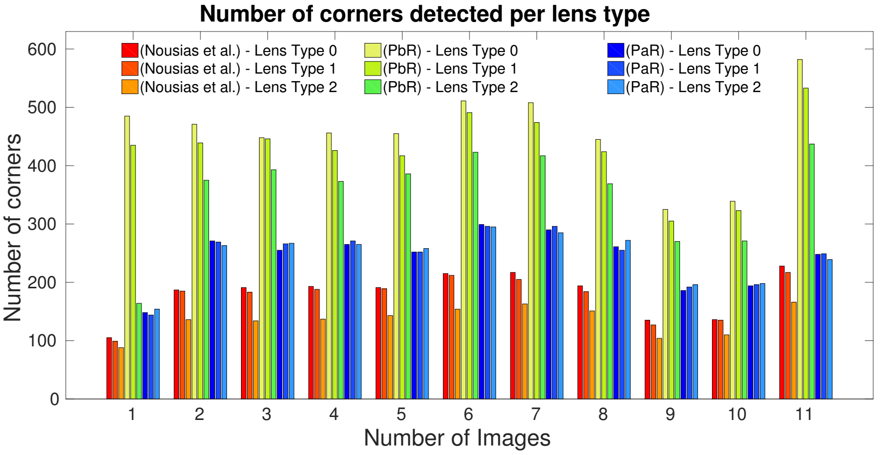


Figure 5.7: Number of corners found for method proposed before (PbR) and after refinement (PaR) step and state of the art method [115].

The performance of the corner detectors is evaluated on the estimation of the synthetic MPC parameters considering three different sets of corners: (i) the ground truth corners provided by the synthetic dataset, and (ii) the corners detected by the algorithm proposed by Nousias *et al.* [115] and (iii) by the proposed algorithm (Section 5.5.2). These corners are used by the proposed calibration procedure and the state of the art calibration procedure for MPCs [115] (denoted as *Nousias17*). For this comparison, is considered the Root Mean Square (RMS) of the reprojection and reconstruction errors for the different stages of the calibration process: the initial linear solution and the nonlinear refinement with and without distortion estimation. The results obtained are summarized in

Tables 5.3 and 5.4.

Reprojection Error [pix]		**Corner Detector**		
Calibration Procedure		Ground Truth	*Nousias17* [115]	Proposed
Initial	*Nousias17* [115]	7.939	4.684	7.887
	*Nousias17** [115]	17.155	3.541	12.402
	Proposed	0.393	2.249	**0.950**
Optimized	*Nousias17* [115]	0.720	1.202	4.273
	*Nousias17** [115]	0.197	0.727	0.534
	Proposed	0.216	0.748	**0.533**
Optimized (with Distortion)	Proposed	0.213	0.743	**0.528**

Table 5.3: RMS reprojection error in pixels for synthetic dataset considering different calibration procedures and corner detectors. The highlighted values correspond to the best result for a given stage of the calibration. * denotes the calibration procedure defined by Nousias *et al.* [115] with the correction in Section 5.4.

Reconstruction Error [mm]		**Corner Detector**		
Calibration Procedure		Ground Truth	*Nousias17* [115]	Proposed
Initial	*Nousias17* [115]	3154.9	1795.6	775.9
	Nousias17 * [115]	133.4	86.8	13.9
	Proposed	2.3	18.9	**5.0**
Optimized	*Nousias17* [115]	52.3	55.6	485.5
	Nousias17 * [115]	39.7	53.9	8.3
	Proposed	1.1	6.6	**5.3**
Optimized (with Distortion)	Proposed	1.4	10.6	**5.7**

Table 5.4: RMS reconstruction error in mm for synthetic dataset considering different calibration procedures and corner detectors. The highlighted values correspond to the best result for a given stage of the calibration. * denotes the calibration procedure defined by Nousias *et al.* [115] with the correction in Section 5.4.

In Tables 5.3 and 5.4, the reprojection and reconstruction errors for the calibration proposed using the ground truth corners attain small values which shows that the camera model defined in Section 5.5.1 is suitable to represent MPCs. The reprojection error is similar to the one obtained using the correction defined in Section 5.4 for the state of the art calibration of Nousias *et al.* [115] while the reconstruction error obtained using the calibration proposed is significantly smaller. One should highlight that the proposed camera model does not need to know the position of the microlenses centers, contrarily to the method of Nousias *et al.* [115].

The correction proposed in Section 5.4 to the calibration procedure of Nousias *et al.* [115] provides better results than applying directly the methodology of Nousias *et al.* [115]. Namely, the reprojection error decreases by 72.6% and the reconstruction error decreases by 24.1%.

Comparing the proposed corner detector with the one proposed by Nousias *et al.* [115], one can see that the proposed corner detector allows to attain smaller reprojection and reconstruction errors in the nonlinear refinement stage. The smallest errors are obtained using the proposed corner detector and the proposed calibration procedure. In this case, the reprojection error attains sub-pixel error in the linear solution and decreases by 43.9% in the nonlinear refinement to 0.53 pixels. On the other hand, the reconstruction error is below 6 mm.

Raytrix Dataset Results. A Raytrix camera is used to obtain images from 10 different poses of a calibration pattern with a 8×6 grid of 48.2×36.2 mm cells. The estimation of the MPC parameters using the calibration procedure proposed is performed using the corners identified with the detector proposed and with the corner detector [115]. The results are compared with the state of the art calibration procedure [115]. As in the synthetic dataset, the results are compared using the reprojection and reconstruction errors for the different stages of the calibration. The results are presented in Table 5.5.

The corners identified using the proposed detector allow to estimate camera parameters that exhibit consistently smaller errors than the ones obtained using the corners identified with the detector [115]. More specifically, the reprojection error reduces by 50.1% and the reconstruction error decreases by 81.3% for the state of the art calibration procedure [115] with the correction defined in Section 5.4. For the calibration proposed, the reprojection error reduces by 34.3% and the reconstruction error decreases by 68.2%. The combination that provides the smallest errors corresponds to the proposed calibration procedure with the proposed corner detector, as in the synthetic dataset. In this case, the reprojection error

Calibration Procedure		**Corner Detector**			
		Reprojection Error [pix]		*Reconstruction Error* [mm]	
		Nousias17 [115]	Proposed	*Nousias17* [115]	Proposed
Initial	*Nousias17* [115]	9.437	4.899	2158.4	3037.5
	*Nousias17** [115]	2.800	**2.287**	200.6	56.3
	Proposed	18.311	4.621	176.9	**14.6**
Optimized	*Nousias17* [115]	4.063	2.178	621.5	899.4
	*Nousias17** [115]	1.165	0.581	245.6	45.9
	Proposed	0.791	**0.520**	10.8	**6.3**
Optimized (with Distortion)	Proposed	0.786	**0.514**	16.3	**8.2**

Table 5.5: RMS reprojection and reconstruction errors for Raytrix dataset considering different calibration procedures and corner detectors. The highlighted values correspond to the best result for a given stage of the calibration. * denotes the calibration procedure defined by Nousias *et al.* [115] with the correction in Section 5.4.

Blur Error [pix]	**Overall**	**Microlens Types**		
Calibration Stage		*Type 1*	*Type 2*	*Type 3*
Initial	0.424	0.422	0.370	0.480
Optimized	0.404	0.353	0.364	0.494
Optimized (with Distortion)	0.401	0.353	0.359	0.493

Table 5.6: RMS blur radius error in pixels for Raytrix dataset for the calibration procedure proposed using the blur radius identified by the detector proposed.

Model	k_u	k_v	u_0	v_0	x_0 [m]	y_0 [m]	z_0 [m]	Δu_0	Δv_0	Δx_0 [mm]	Δy_0 [mm]
Proposed	2352.30	2336.40	-257.92	-230.20	0.13	0.11	-1.16	0.90	1.54	0.44	0.77
Bok *et al.* [21]	2535.38	2535.38	-144.24	-208.09	0.07	0.10	-1.21	0.94	1.64	0.45	0.78
Ratio	1.08	1.09	0.56	0.90	0.54	0.87	1.04	1.05	1.06	1.01	1.02

Table 5.7: Parameters of the camera model proposed in Section 5.1. The parameters are estimated using the calibration procedure proposed (Section 5.5.3) and by transforming the parameters (camera model equivalent parameters) estimated using the calibration procedure of Nousias *et al.* [115] according with the mappings defined in Section 5.4. The ratio between the camera model equivalent parameters and the estimated using the proposed calibration is presented in the last row.

decreases to 0.52 pixels (10.5% decrease) and the reconstruction error decreases to 6.3 mm (86.3% decrease).

The radial distortion present in the Raytrix images acquired is very small, therefore the decrease in the reprojection error with the estimation of radial distortion is only 1.2%. The previous discussions did not include the radial distortion because the state of the art method does not consider distortion estimation during the calibration.

The results in the synthetic and Raytrix dataset show that besides the common extrinsic parameters [115], the MPC can be described using common intrinsic parameters among the microlens types. More specifically, considering an intrinsic model with 7 parameters (Section 5.3), one is able to obtain smaller or similar reprojection and reconstruction errors with the state of the art calibration procedure [115] that considers 6 parameters for each microlens type in a total of 18 parameters.

Blur Model. Table 5.6 presents the blur radius error obtained with the blur model (5.24) used to represent the different microlens types. The blur model parameters are estimated using the calibration procedure and detector proposed. The overall blur error obtained after nonlinear refinement is 0.40 pixels. More specifically, the blur error for the microlens type 1 is 0.35 pixels, for the microlens type 2 is 0.36 pixels and for the microlens type 3 is 0.49 pixels. The sub-pixel blur radius error shows that the blur model is suitable to represent the defocus exhibited by the different microlens types. In Figure 5.8, one can see that the blur radius estimated is in accordance with the blur radius detected. The blur model gives us a different focal plane depth for each microlens type as expected. Namely, there are two microlenses (types 1 and 2) focusing at depths near the camera (1.19 m and 1.53 m) and one microlens (type 3) focusing at a depth farther away from the camera (2.66 m).

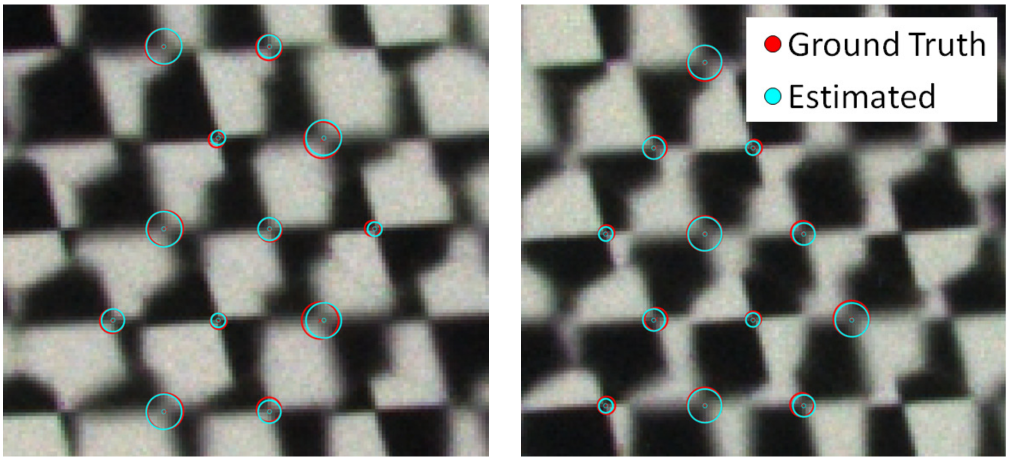


Figure 5.8: Examples of the blur radius estimated using the calibration proposed (cyan circles) and comparison with the detected blur radius (red circles).

Bok et al. *[21] Comparison.* In Table 5.7, one presents the parameters obtained for the camera model proposed (Section 5.1). The parameters are either estimated using the calibration procedure proposed (Section 5.5.3) or by transforming the camera model parameters (denoted as camera model equivalent parameters) of Bok *et al.* [21] according with the mappings defined in Section 5.4.

The camera model parameters of Bok *et al.* [21] are obtained using the calibration procedure of Nousias *et al.* [115] and considering the same detected corner points as the ones used in the calibration proposed. In the calibration procedure of Nousias *et al.* [115], one assumes additionally that there is only one microlens type as considered for modeling the MPC. The calibration results in the following additional intrinsic parameters $K_1 = 18.48$ and $K_2 = -22291.00$, and coordinates for converting normalized coordinates to image coordinates $\left(f_x, f_y, c_x, c_y\right) = (46853.00, 46853.00, 2682.10, 3858.00)$.

Additionally, for transforming the Bok *et al.* [21] parameters to the camera model equivalent parameters, one needs to know the origin (p_0, g_0) and the spacing (d_h, d_v) between microlenses. These parameters are obtained by analyzing the microlens centers in the white image during the process of defining the microlens coordinates (k, l). This analysis gives an horizontal distance of $d_h = 34.89$ pixels and a vertical distance of $d_v = 30.22$ pixels with an origin defined by $(p_0, g_0) = (16.50, 12.63)$ pixels.

The camera model parameters estimated and the camera model equivalent parameters are very similar. Namely, most of the parameters are within a maximum deviation of 10%. The exceptions correspond to the principal point u_0 and the (x_0, y_0) coordinates for the origin of the camera coordinate system. The different estimates for these parameters can be caused by the different calibration procedures used and in part can explain the different results in terms of reprojection and reconstruction errors.

5.6 Chapter Summary

In this chapter was defined the mapping between the LFIM and the microlens cameras, and one establishes the equivalence between the projection set $\mathcal{P}_{ij}$ (3.17) and the projections of the microlens cameras. The mapping between these models was used to establish a relationship between the LFIM and the microlens camera model of Bok *et al.* [21]. In Section 5.4 was shown that the model proposed by Bok *et al.* [21] is equivalent to the LFIM constraining the microlenses to be regularly spaced.

The microlens cameras define a coplanar array of cameras that differ on the location of their projection centers and on their principal points (Section 5.1.2). The different principal points define an EPI geometry whose zero disparity plane is at the plane containing the viewpoint projection centers (Section 5.1.3).

The pinhole microlens camera constraint (5.6) allows to represent the LFIM introduced in [41] using eight free intrinsic parameters. This is accomplished by shifting the rays parameterization plane along the optical axis of the camera [17] to the plane containing the microlens projection centers and removing redundant parameters with the extrinsic parameters (Section 5.2).

The microlens camera array model derived can be represented with a minimum of 6 parameters (Section 5.3) by shifting the rays parameterization plane along the optical axis of the camera [17] to the plane containing the viewpoint projection centers, removing the redundant parameters with the extrinsic parameters, and assuming the pinhole microlens camera constraint and that the depth of a point estimated through the EPIs is the same.

The microlens camera array model complemented by the blur model [11, 18] is used to define a linear solution for the MPC that considers three steps: i) a Direct Linear Transformation (DLT) calibration to obtain the parameters that describe the microlens parametric homography matrix from point correspondences $(\tilde{\mathbf{m}}, \tilde{\mathbf{q}})$, (ii) a strategy to decompose this homography matrix into intrinsic and extrinsic parameters based on a parametric representation of the image of the absolute conic, (iii) and a method to retrieve the blur model associated with each microlens type. The calibration procedure proposed for the MPC is based on corner points and blur radius detected in the MIs using a new detection algorithm that overcomes the defocus blur present in the MIs. The corner detection algorithm and the calibration proposed outperform the state of the art showing that the MPC can be described using common intrinsic and extrinsic parameters among the different microlens types (Section 5.5).

The next chapter will define the geometry associated with other virtual cameras that can be obtained from the capture Lightfield (LF) and will present an extension of the LFIM to arrays of camera arrays.

Chapter 6

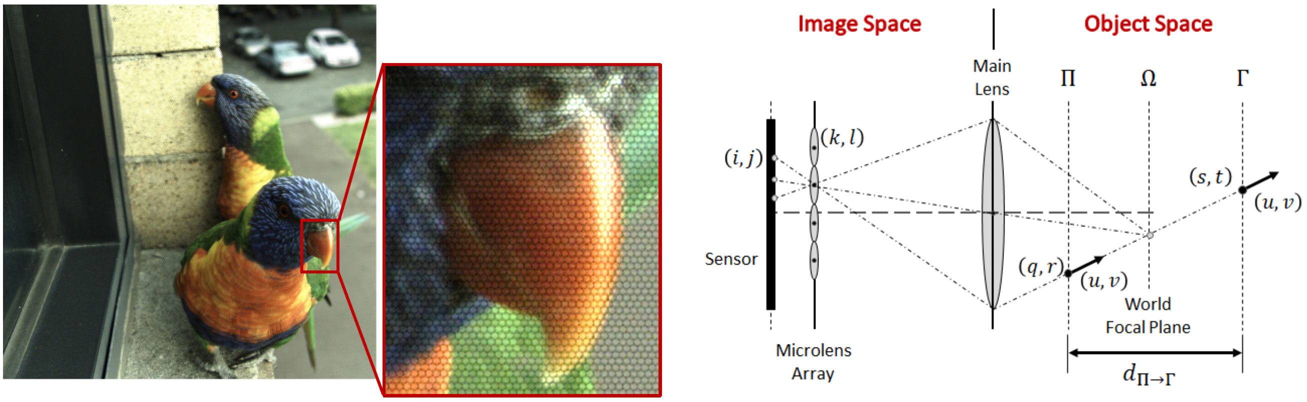
Depth-Selected Camera Arrays

The Lightfield Intrinsic Matrix (LFIM) allows to represent a coplanar camera array of viewpoints (Section 4.1) and microlenses (Section 5.1). These arrays are obtained either fixing the pixels (i, j) or the microlenses (k, l). In this chapter are explored the different combinations of rays captured by a plenoptic camera to fully formalize the corresponding camera arrays. In addition, is described the geometry of a coplanar plenoptic camera array to propose an extension of the LFIM.

6.1 Camera Array Redefinition

The Lightfield (LF) captured by Standard Plenoptic Cameras (SPCs) can be represented using several types of images by reorganizing the pixels captured by the camera on the 2D raw image [112]. The raw image displays the pixels collected by each microlens in the microlens array (Figure 6.1.a) and represents the images captured by the physical microlens array placed in front of the sensor (Section 5.1). There is another arrangement of pixels that is commonly used in SPCs, the Viewpoint Images (VIs). These images are obtained by selecting the same pixel position relatively to the microlens center for each microlens [112]. This rearrangement defines a virtual camera array with coplanar projection centers and with a very narrow baseline [104] (Section 4.1).

The rays captured by an SPC allow defining alternative cameras with rays intersecting at an arbitrary point (depth) in the scene [103] either by applying a shearing operation or creating Surface Camera Images



(a) (b)

Figure 6.1: Raw image and ray parameterization in an SPC. **(a)** Raw image captured by an SPC with detail of the hexagonal microlens array tiling. **(b)** The LF in the image space is parameterized using pixels and microlenses indices while the LF in the object space is parameterized using a point and a direction.

(SCams). Although these strategies are commonly used for disparity estimation [29, 134], the geometry associated with the corresponding cameras has not been defined. In this section, is fully formalized the geometry of the multiple viewpoint and microlens camera arrays that can be obtained and are derived the corresponding mappings from the LFIM **H** used to model an SPC.

6.1.1 Generalized Camera Arrays

In this section, is shown that one plenoptic camera can define multiple camera arrays. The multiplicity comes from the fact that one may collect rays with different combinations of pixel and microlens coordinates. Let us start by defining the possible combinations of pixel and microlens coordinates.

Surface Camera Images and Shearing. A SCam (Section 2.5.3) collects rays that intersect at an arbitrary point in the object space. Considering the LF in the object space $L_\Pi(q,r,u,v)$ acquired by a plenoptic camera (Figure 6.1.b) described by the LFIM $\mathbf{H}_\Pi$ (4.7), one can obtain a SCam with projection center at point (s,t) of plane Γ at a distance $d_{\Pi\to\Gamma}$ of plane Π using the re-parameterization of the LF (Section 3.3.1).

The re-parameterization (3.3) allows to define a constraint to identify the rays of the LF in the image space $L(i, j, k, l)$ that intersect at an arbitrary point of the plane Γ [103]. Let $\mathbf{\Phi}_a$ and $\mathbf{\Phi}_b$ be two rays in the image space with the same coordinates (s, t) on plane Γ, by taking their difference one defines a constraint on the LF coordinates in the image space as

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} = \mathbf{H}_{ij}^{st} \begin{bmatrix} \Delta i \\ \Delta j \end{bmatrix} + \mathbf{H}_{kl}^{st} \begin{bmatrix} \Delta k \\ \Delta l \end{bmatrix} \tag{6.1}$$

where $\Delta(\cdot) = (\cdot)_b - (\cdot)_a$, and $\mathbf{H}_{(\cdot)}^{st}$ corresponds to 2×2 sub-matrices of $\mathbf{H}_\Gamma$ obtained from selecting the entries of the first two rows, denoted by st, and selecting either the entries of the 1^{st} and 2^{nd} columns, denoted by ij, or the 3^{rd} and 4^{th} columns, denoted by kl. The LFIM $\mathbf{H}_\Gamma$ maps the rays in the image space $\mathbf{\Phi} = [i, j, k, l]^T$ to the rays in the object space $\mathbf{\Psi}_\Gamma = [s, t, u, v]^T$. Using the constraint (6.1) and considering (i_r, j_r, k_r, l_r) as reference coordinates to enforce the constraint, one can define the set of rays that compose a SCam considering a sampling fixing the viewpoint coordinates (i, j) as

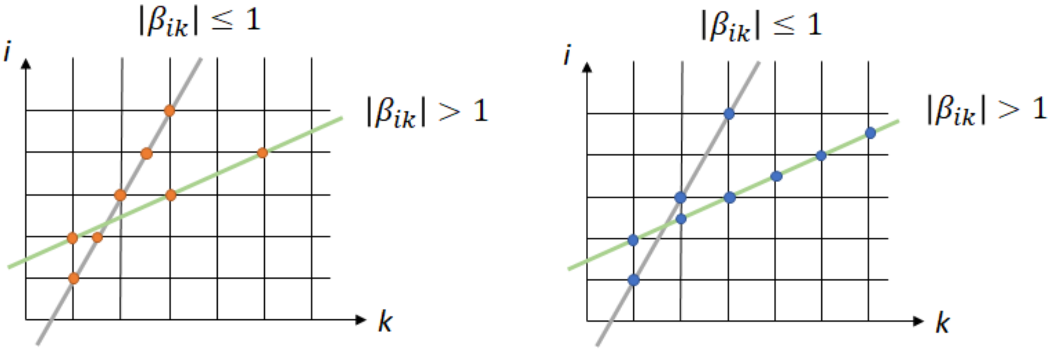
$$k = k_r + \beta_{ik}(i - i_r) \wedge l = l_r + \beta_{jl}(j - j_r) \tag{6.2}$$

where the parameters $\beta_{ik} = -\frac{h_{qi} + d_{\Pi\to\Gamma}\, h_{ui}}{h_{qk} + d_{\Pi\to\Gamma}\, h_{uk}}$ and $\beta_{jl} = -\frac{h_{rj} + d_{\Pi\to\Gamma}\, h_{vj}}{h_{rl} + d_{\Pi\to\Gamma}\, h_{vl}}$ correspond to the disparities considered on the VIs for a point at depth $d_{\Pi\to\Gamma}$ (3.15). Alternatively, one can define the set of rays considering a sampling fixing the microlens coordinates (k, l) as

$$i = i_r + \beta_{ik}^{-1}(k - k_r) \wedge j = j_r + \beta_{jl}^{-1}(l - l_r) \tag{6.3}$$

where the parameters β_{ik}^{-1} and β_{jl}^{-1} correspond to the disparities considered on the Microlens Images (MIs) for a point at depth $d_{\Pi\to\Gamma}$. Assuming that one wants to maximize the number of sampled rays, one should use (6.2) for absolute disparities lower or equal than one ($|\beta_{ik}| \leq 1$ or

$\left|\beta_{jl}\right| \leq 1$) and (6.3) for absolute disparities greater than one ($|\beta_{ik}| > 1$ or $\left|\beta_{jl}\right| > 1$) (Figure 6.2).



(a)

(b)

Figure 6.2: Sampling of the LF fixing the viewpoint coordinates **(a)** or fixing the microlens coordinates **(b)**. The sampling fixing the viewpoint coordinates does not allow to maximize the LF sampling for $|\beta_{ik}| > 1$ while the sampling fixing the microlens coordinates does not allow to maximize the LF sampling for $|\beta_{ik}| \leq 1$.

The sampling (6.2) is associated with the sampling performed during the shearing operation defined by Tao *et al.* [134]. Shearing (Section 2.5.4) can be interpreted as a resampling of the acquired LF in order to have the rays that intersect at an arbitrary point (s, t) of the plane Γ in the same virtual microlens (k_s, l_s), *i.e.* the rays collected in each microlens of the sheared LF corresponds to a SCam whose projection center lies on the plane Γ. Considering the sampling (6.2), the rays are mapped to the same microlens if $(k_s, l_s) = (k_r, l_r)$. Assuming that $\beta_{ik} = \beta_{jl} = \beta$ and denoting the rays in the sheared LF as $\mathbf{\Phi}_s = [i, j, k_s, l_s]^T$, the relationship between the rays of the acquired and the sheared LF can be defined as

$$\tilde{\mathbf{\Phi}}_s = \mathbf{U}_{kl}\tilde{\mathbf{\Phi}} \tag{6.4}$$

where

$$\mathbf{U}_{kl} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ -\beta & 0 & 1 & 0 & \beta\, i_r \\ 0 & -\beta & 0 & 1 & \beta\, j_r \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} . \tag{6.5}$$

On the other hand, the sampling (6.3) can be interpreted as a resampling of the LF $L\left(i,j,k,l\right)$ such that the rays that intersect at an arbitrary point (s,t) of the plane Γ are collected in the same virtual viewpoint (i_s,j_s), *i.e.* each viewpoint corresponds to a SCam. Considering the sampling (6.3), the rays are mapped to the same viewpoint if $(i_s,j_s) = (i_r,j_r)$. Assuming that $\beta_{ik} = \beta_{jl} = \beta$ and denoting the rays in the resampled LF as $\mathbf{\Phi}_s = \left[i_s, j_s, k, l\right]^T$, the relationship between the rays of the acquired and the resampled LF can be defined as

$$\tilde{\mathbf{\Phi}}_s = \mathbf{U}_{ij}\tilde{\mathbf{\Phi}} \tag{6.6}$$

where

$$\mathbf{U}_{ij} = \begin{bmatrix} 1 & 0 & -\beta^{-1} & 0 & \beta^{-1}\, k_r \\ 0 & 1 & 0 & -\beta^{-1} & \beta^{-1}\, l_r \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} . \tag{6.7}$$

The sampling of rays in the acquired LF to define a SCam is defined either by fixing the viewpoint or the microlens coordinates. Thus, let us now fully formalize the projection model associated with the different types of cameras defined using the two sampling approaches.

Sampling Fixing Viewpoint Coordinates. Let us start considering the sampling fixing the viewpoint coordinates (6.2). This sampling is the one associated with the LF shearing operation. Consider also the LF in the object space whose rays are parameterized at a plane Π using a point $[q, r, 0]^T$ and a direction $[u, v, 1]^T$ (Figure 6.1.b). The LFIM

$$\mathbf{H}_{kl} = \mathbf{H}_{\Pi}\mathbf{U}_{kl}^{-1} = \begin{bmatrix} h_{qi} + \beta\, h_{qk} & 0 & h_{qk} & 0 & h_q - \beta\, h_{qk}\, i_r \\ 0 & h_{rj} + \beta\, h_{rl} & 0 & h_{rl} & h_r - \beta\, h_{rl}\, j_r \\ h_{ui} + \beta\, h_{uk} & 0 & h_{uk} & 0 & h_u - \beta\, h_{uk}\, i_r \\ 0 & h_{vj} + \beta\, h_{vl} & 0 & h_{vl} & h_v - \beta\, h_{vl}\, j_r \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.8}$$

maps the rays in the image space $\mathbf{\Phi}_s$ that define a SCam with projection center at plane Γ to the rays in the object space $\mathbf{\Psi}_{\Pi}$.

The caustic surfaces and the constraints defined in Sections 4.1.2 and 5.1.2 are generic for any LFIM $\mathbf{H}$. Thus, let us define the projection centers associated with the resampled viewpoint cameras (i, j) and the resampled microlens cameras (k_s, l_s).

Viewpoint Projection Matrix. Replacing the LFIM entries in (4.10) by the corresponding entries of $\mathbf{H}_{kl}$ (6.8), one concludes that the solution for the vanishing constraint is the same. Therefore, the caustic profile for the resampled viewpoint camera (i, j), the constraint to ensure a unique projection center and the location of the projection center are the same as the ones defined in Section 4.1.2.

The projection matrix associated with the viewpoint that results from the sampling fixing the viewpoint coordinates is obtained considering the unique viewpoint projection center constraint (4.11) and solving the back-projection (3.11) redefined with the LFIM $\mathbf{H}_{kl}$ (6.8) relatively to (k_s, l_s). Rewriting the resulting equation as a pinhole model like (4.1), one obtains an intrinsic matrix defined as

$$\mathbf{K}^{ij} = \begin{bmatrix} \frac{1}{h_{uk}} & 0 & -\frac{h_u}{h_{uk}} - i\,\frac{h_{ui}}{h_{uk}} - \beta\,(i - i_r) \\ 0 & \frac{1}{h_{vl}} & -\frac{h_v}{h_{vl}} - j\,\frac{h_{vj}}{h_{vl}} - \beta\,(j - j_r) \\ 0 & 0 & 1 \end{bmatrix} \tag{6.9}$$

and the projection center as $\mathbf{t}^{ij} = -\mathbf{p}_c$ (4.12). More in detail, the camera model for the resampled viewpoint camera only differs on the principal point relatively to the viewpoint camera (4.13) (Figure 6.3.b-c), which is consistent with the strategy to translate the VIs to perform shearing of the LF [76, 134]. Thus, the mapping with (4.2) only differs in the common component $[u_0, v_0]^T = -\left[h_u/h_{uk} - \beta\, i_r,\; h_v/h_{vl} - \beta\, j_r\right]^T$ and in the differential component $[\Delta u_0, \Delta v_0]^T = -\left[h_{ui}/h_{uk} + \beta,\; h_{vj}/h_{vl} + \beta\right]^T$ of the principal point relatively to the mapping presented in Section 4.1.2.

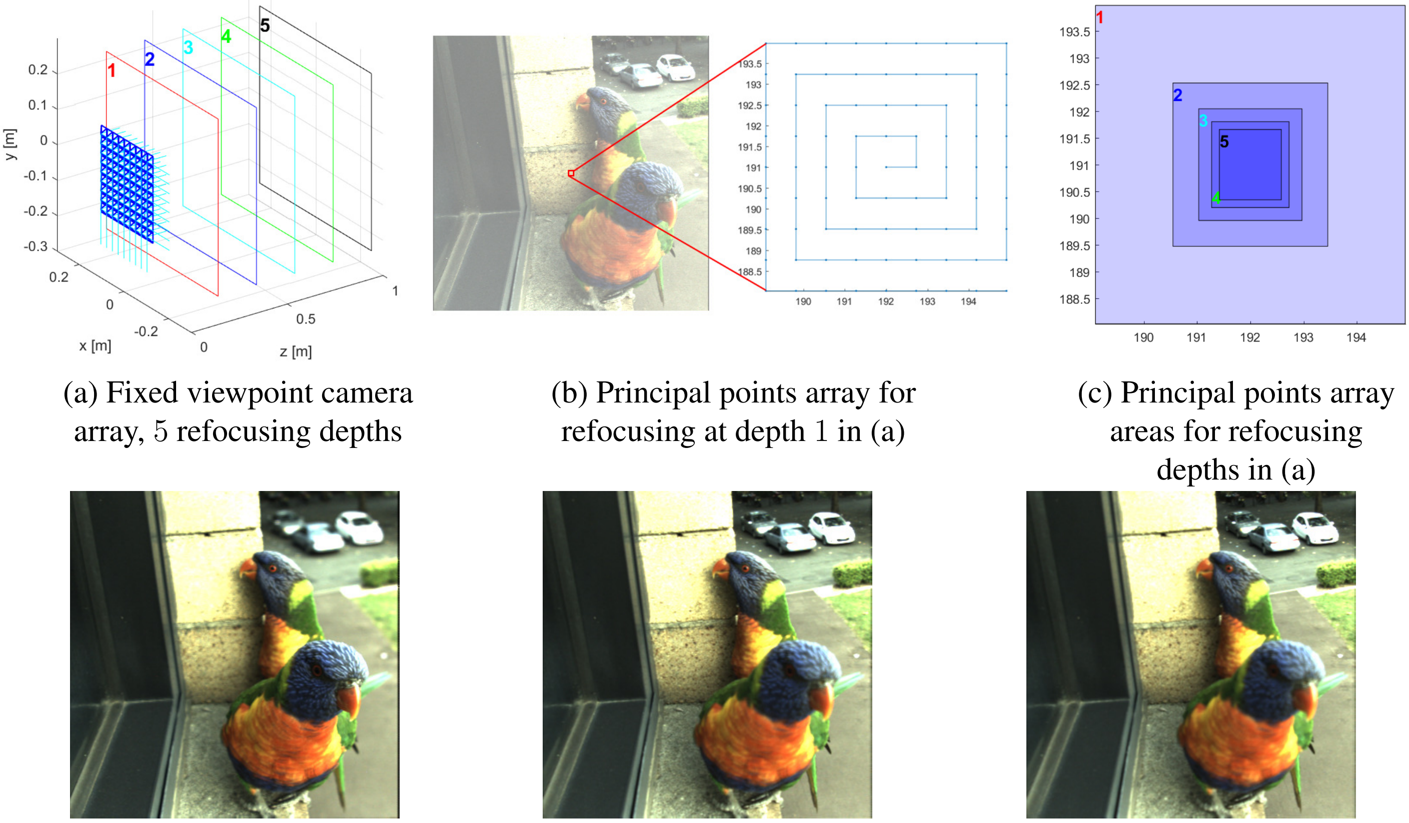
Microlens Projection Matrix. Replacing the LFIM entries in (5.5) by the corresponding entries of $\mathbf{H}_{kl}$ (6.8), one concludes that the solution for the vanishing constraint changes and is given by:

$$\lambda_1 = -\frac{h_{qi} + \beta\, h_{qk}}{h_{ui} + \beta\, h_{uk}} \quad \vee \quad \lambda_2 = -\frac{h_{rj} + \beta\, h_{rl}}{h_{vj} + \beta\, h_{vl}} \quad . \tag{6.10}$$

This results in a caustic profile for the resampled microlens camera (k_s, l_s) that is similar to the one presented in Section 5.1.2 but with different spacing in the x- and y- dimensions and with a different z coordinate. Consequently, the constraint to ensure a unique projection center for the microlens camera (k_s, l_s) also changes:

$$\frac{h_{qi} + \beta\, h_{qk}}{h_{ui} + \beta\, h_{uk}} = \frac{h_{rj} + \beta\, h_{rl}}{h_{vj} + \beta\, h_{vl}} \quad , \tag{6.11}$$

and the projection center is defined on a plane at a depth $z_\beta = -\frac{h_{qi} + \beta\, h_{qk}}{h_{ui} + \beta\, h_{uk}}$ by



(a) Fixed viewpoint camera array, 5 refocusing depths

(b) Principal points array for refocusing at depth 1 in (a)

(c) Principal points array areas for refocusing depths in (a)

(d) Refocusing at depth 1, faraway cars blurred

(e) Refocusing at depth 3, bricks texture focused

(f) Refocusing at depth 5, nearest parrot blurred

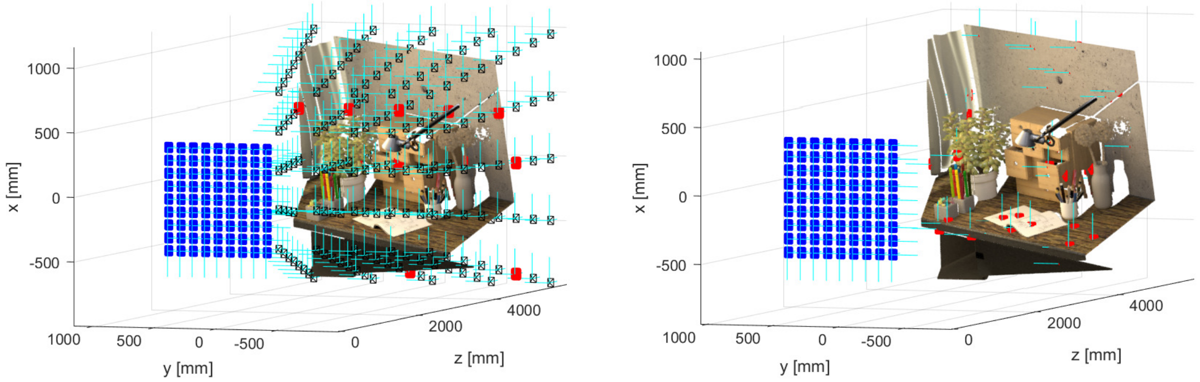
Figure 6.3: Viewpoint camera arrays obtained considering shearing for refocusing at depths $z = 0.2, 0.4, \ldots, 1.0$ m **(a)**. The spacing among projection centers has been scaled 100 times to be perceptible on the 3D plot. The distribution of the principal points for the viewpoint camera arrays at different refocusing depths are depicted in **(b)** and **(c)**. The corresponding refocused images are depicted in **(d)**, **(e)** and **(f)**.

$$\mathbf{p}_c = \begin{bmatrix} h_q + z_\beta\, h_u + \big(h_{qk} + z_\beta\, h_{uk}\big)\,(k_s - \beta\, i_r) \\ h_r + z_\beta\, h_v + \big(h_{rl} + z_\beta\, h_{vl}\big)\,(l_s - \beta\, j_r) \\ z_\beta \end{bmatrix} \,. \tag{6.12}$$

The projection matrix associated with the resampled microlens is obtained considering the unique projection center constraint (6.11) and solving the back-projection (3.11) redefined with the LFIM $\mathbf{H}_{kl}$ (6.8) relatively to (i, j). Rewriting the resulting equation as a pinhole model like (5.1), one obtains an intrinsic matrix defined as

$$\mathbf{K}^{kl} = \begin{bmatrix} \frac{1}{h_{ui}+\beta\, h_{uk}} & 0 & -\frac{h_u-\beta\, h_{uk}\, i_r}{h_{ui}+\beta\, h_{uk}} - k_s\, \frac{h_{uk}}{h_{ui}+\beta\, h_{uk}} \\ 0 & \frac{1}{h_{vj}+\beta\, h_{vl}} & -\frac{h_v-\beta\, h_{vl}\, j_r}{h_{vj}+\beta\, h_{vl}} - l_s\, \frac{h_{vl}}{h_{vj}+\beta\, h_{vl}} \\ 0 & 0 & 1 \end{bmatrix} \tag{6.13}$$

and the projection center as $\mathbf{t}^{kl} = -\mathbf{p}_c$ (6.12). The camera model of the resampled microlens camera is completely different from the microlens camera (5.8). Namely, comparing (6.13) with (5.2), one identifies the scale factors as $k_u = \frac{1}{h_{ui}+\beta h_{uk}}$ and $k_v = \frac{1}{h_{vj}+\beta h_{vl}}$. The common component of the principal point is defined as $[u_0, v_0]^T = -\Big[k_u\,(h_u - \beta\; h_{uk}\; i_r)\,,\;\; k_v\,(h_v$ and the differential component as $[\Delta u_0, \Delta v_0]^T = -\,[k_u\; h_{uk},\;\; k_v\; h_{vl}]^T$. The baseline is defined as $[\Delta x_0, \Delta y_0, 0]^T = -[h_{qk}+z_\beta\; h_{uk},\;\; h_{rl}+z_\beta\; h_{vl},\;\; 0]^T$. Finally, the location of the resampled microlens camera array relatively to the camera coordinate system origin is defined as $[x_0, y_0, z_0]^T = -\Big[h_q + z_\beta\; h_u$



(a) Microlens camera arrays for different refocused depths

(b) Microlens cameras corresponding to surface points (SCam)

Figure 6.4: Microlens camera arrays obtained considering shearing for refocusing at different depths for the synthetic Table dataset [68]. Shearing allows to obtain microlens cameras with projection centers at different depths **(a)**. These cameras obtain relevant information for depth estimation [29] when the projection center corresponds to a surface point, *i.e.* a SCam is defined **(b)**. The viewpoint camera array is represented in blue with the spacing among projection centers scaled by 4 times.

Notice that replacing $\beta = -\frac{h_{qi}+\; d_{\Pi\to\Gamma}\; h_{ui}}{h_{qk}+\; d_{\Pi\to\Gamma}\; h_{uk}}$ in z_β, one can see that the depth of the projection center for the resampled microlens camera (k_s, l_s) corresponds to the plane Γ at $d_{\Pi\to\Gamma}$ (Figure 6.4).

The SCam considering the sampling fixing the viewpoint coordinates corresponds to a microlens camera in the resampled LF. Therefore, the change that occurs only in the microlens camera projection center is in accordance.

Generalized Epipolar Plane Image (EPI) Geometry. Considering equation (3.11) and the resampled viewpoint cameras (6.9), one can obtain the EPI geometry that relates the depth of a point with the disparity on the VIs $\left[\frac{\Delta k_s}{\Delta i}, \frac{\Delta l_s}{\Delta j}\right]^T$ as

$$\frac{\Delta k_s}{\Delta i} = -\frac{h_{qi} - \frac{h_{qk}}{h_{uk}} h_{ui}}{h_{uk}} \frac{1}{z + \frac{h_{qk}}{h_{uk}}} - \frac{h_{ui}}{h_{uk}} - \beta \text{ and } \frac{\Delta l_s}{\Delta j} = -\frac{h_{rj} - \frac{h_{rl}}{h_{vl}} h_{vj}}{h_{vl}} \frac{1}{z + \frac{h_{rl}}{h_{vl}}} - \frac{h_{vj}}{h_{vl}} - \tag{6.14}$$

The EPI geometry shows that the zero disparity plane, also known as the optical focal plane [112] of the main lens is affected by the shearing operation. This is in accordance with the creation of a virtual focal plane during the refocus operation that implicitly requires a shearing of the LF [112] (Figure 6.3.d-f).

Sampling Fixing Microlens Coordinates. Now, let us consider the sampling fixing the microlens coordinates (6.3). Consider also the LF in the object space whose rays are parameterized at a plane Π using a point $[q, r, 0]^T$ and a direction $[u, v, 1]^T$ (Figure 6.1.b). The LFIM

$$\mathbf{H}_{ij} = \mathbf{H}_{\Pi}\mathbf{U}_{ij}^{-1} = \begin{bmatrix} h_{qi} & 0 & h_{qk} + \beta^{-1} h_{qi} & 0 & h_q - \beta^{-1} h_{qi} k_r \\ 0 & h_{rj} & 0 & h_{rl} + \beta^{-1} h_{rj} & h_r - \beta^{-1} h_{rj} l_r \\ h_{ui} & 0 & h_{uk} + \beta^{-1} h_{ui} & 0 & h_u - \beta^{-1} h_{ui} k_r \\ 0 & h_{vj} & 0 & h_{vl} + \beta^{-1} h_{vj} & h_v - \beta^{-1} h_{vj} l_r \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.15}$$

maps the rays in the image space $\mathbf{\Phi}_s$ that define a SCam with projection center at plane Γ to the rays in the object space $\mathbf{\Psi}_{\Pi}$.

Viewpoint Projection Matrix. Similarly to the sampling fixing the viewpoint coordinates, let us replace the LFIM entries in (4.10) by the corresponding entries of $\mathbf{H}_{ij}$ (6.15). This allows to conclude that the solution for the vanishing constraint changes and is the same as the one obtained for the resampled microlens camera (6.10). Therefore, the caustic profile is similar to the one for the resampled microlens camera. Namely, it has different x- and y- spacings but has the same z coordinate. Additionally, the constraint to ensure a unique projection center is the same as the one obtained for the resampled microlens camera (6.11) and the projection center is defined as

$$\mathbf{p}_c = \begin{bmatrix} h_q + z_\beta\, h_u + \left(h_{qi} + z_\beta\, h_{ui}\right) \left(i_s - \beta^{-1}\, k_r\right) \\ h_r + z_\beta\, h_v + \left(h_{rj} + z_\beta\, h_{vj}\right) \left(j_s - \beta^{-1}\, l_r\right) \\ z_\beta \end{bmatrix} \quad , \tag{6.16}$$

where $z_\beta = -\frac{h_{qi} + \beta\, h_{qk}}{h_{ui} + \beta\, h_{uk}}$.

The projection matrix associated with the resampled viewpoint is obtained considering the unique projection center constraint (6.11) and solving the back-projection (3.11) redefined with the LFIM $\mathbf{H}_{ij}$ (6.15) relatively to (k, l). Rewriting the resulting equation as a pinhole model like (4.1), one obtains an intrinsic matrix defined as

$$\mathbf{K}^{ij} = \begin{bmatrix} \frac{\beta}{h_{ui}+\beta\, h_{uk}} & 0 & -\frac{\beta h_u - h_{ui}\, k_r}{h_{ui}+\beta\, h_{uk}} - i_s\, \frac{\beta h_{ui}}{h_{ui}+\beta\, h_{uk}} \\ 0 & \frac{\beta}{h_{vj}+\beta\, h_{vl}} & -\frac{\beta h_v + h_{vj}\, l_r}{h_{vj}+\beta\, h_{vl}} - j_s\, \frac{\beta h_{vj}}{h_{vj}+\beta\, h_{vl}} \\ 0 & 0 & 1 \end{bmatrix} \tag{6.17}$$

and the projection center as $\mathbf{t}^{ij} = -\mathbf{p}_c$ (6.16). The camera model of the resampled viewpoint camera (i_s, j_s) is different from the viewpoint camera (4.13) and from the resampled microlens camera (6.13). Namely, comparing (6.17) with (4.2), one identifies the scale factors as $k_u = \frac{\beta}{h_{ui}+\beta h_{uk}}$ and $k_v = \frac{\beta}{h_{vj}+\beta h_{vl}}$. The common component of the principal point is defined as $[u_0, v_0]^T = -[k_u(h_u{+}\beta^{-1}\, h_{ui}\, k_r),\ k_v(h_v{+}\beta^{-1}\, h_{vj}\, l_r)]^T$ and the differential component as $[\Delta u_0, \Delta v_0]^T = -[k_u\, h_{ui},\ k_v\, h_{vj}]^T$. The baseline is defined as $[\Delta x_0, \Delta y_0, 0]^T = -[h_{qi} + z_\beta\, h_{ui},\ h_{rj} + z_\beta\, h_{vj},\ 0]^T$. Finally, the location of the resampled viewpoint camera array relatively to the camera coordinate system origin is defined as $[x_0, y_0, z_0]^T = -[h_q{+} z_\beta\, h_u{-}\beta^{-1}\, k_r \left(h_{qi} + z_\beta\, h_{ui}\right),\ h_r{+}z_\beta\, h_v{-}\beta^{-1}\, l_r \left(h_{rj} + z_\beta\, h_{vj}\right),\ -z_\beta]^T$.

Microlens Projection Matrix. Replacing the LFIM entries in (5.5) by the corresponding entries of $\mathbf{H}_{ij}$ (6.15), one concludes that the solution for the vanishing constraint is the same. Therefore, the caustic profile for the microlens camera (k, l), the constraint to ensure a unique projection center and the location of the projection center are the same as the ones defined in Section 5.1.2.

The projection matrix associated with the microlens that results from the sampling fixing the microlens coordinates is obtained considering the unique microlens projection center constraint (5.6) and solving the back-projection (3.11) redefined with the LFIM $\mathbf{H}_{ij}$ (6.15) relatively to (i_s, j_s). Rewriting the resulting equation as a pinhole model like (5.1), one obtains an intrinsic matrix defined as

$$\mathbf{K}^{kl} = \begin{bmatrix} \frac{1}{h_{ui}} & 0 & -\frac{h_u}{h_{ui}} - k\,\frac{h_{uk}}{h_{ui}} - \beta^{-1}\,(k - k_r) \\ 0 & \frac{1}{h_{vj}} & -\frac{h_v}{h_{vj}} - l\,\frac{h_{vl}}{h_{vj}} - \beta^{-1}\,(l - l_r) \\ 0 & 0 & 1 \end{bmatrix} \tag{6.18}$$

and the projection center as $\mathbf{t}^{kl} = -\mathbf{p}_c$ (5.7). As for the resampled viewpoint camera (6.9), the camera model for the resampled microlens camera only differs on the principal point relatively to the microlens camera (5.8). Thus, the mapping with (5.2) only differs in the common component $[u_0, v_0]^T = -[h_u/h_{ui} - \beta^{-1}\, k_r,\ h_v/h_{vj} - \beta^{-1}\, l_r]^T$ and in the differential component $[\Delta u_0, \Delta v_0]^T = -[h_{uk}/h_{ui} + \beta^{-1},\ h_{vl}/h_{vj} + \beta^{-1}]^T$ of the principal point relatively to the mapping presented in Section 5.1.2.

The SCam considering the sampling fixing the microlens coordinates corresponds to a viewpoint camera in the LF. Therefore, the change that occurs only in the viewpoint camera projection center and the same z coordinate between the viewpoint (i_s, j_s) and the microlens (k_s, l_s) projection centers are in accordance.

Generalized EPI Geometry. Considering equation (3.11) and the resampled microlens cameras (6.18), one can obtain the EPI geometry that relates the depth of a point with the disparity on the MIs $\left[\frac{\Delta i_s}{\Delta k}, \frac{\Delta j_s}{\Delta l}\right]^T$ as

$$\frac{\Delta i_s}{\Delta k} = -\frac{h_{qk} - \frac{h_{qi}}{h_{ui}} h_{uk}}{h_{ui}} \frac{1}{z + \frac{h_{qi}}{h_{ui}}} - \frac{h_{uk}}{h_{ui}} - \frac{1}{\beta} \text{ and } \frac{\Delta j_s}{\Delta l} = -\frac{h_{rl} - \frac{h_{rj}}{h_{vj}} h_{vl}}{h_{vj}} \frac{1}{z + \frac{h_{rj}}{h_{vj}}} - \frac{h_{vl}}{h_{vj}} - \tag{6.19}$$

This EPI geometry shows that the zero disparity plane also changes with the sampling of the viewpoint coordinates, similarly to the EPI geometry (6.14). Hence, this sampling can also be used during the refocusing operation of the LF.

6.1.2 Experimental Results

In this section, the mappings proposed in Section 6.1.1 are validated experimentally using the publicly available calibration dataset [41] (Dataset A) acquired with a 1^{st} generation Lytro camera. Namely, the viewpoint and microlens cameras obtained after calibration of the sheared versions of the LFs for the calibration dataset are compared with the cameras obtained using the mappings proposed with the LFIM obtained from the calibration of the non-sheared calibration dataset.

Let us start by calibrating the non-sheared calibration dataset using the calibration procedure described in Section 4.3. The estimated LFIM $\mathbf{H}$ (4.17) and the corresponding viewpoint (4.1) and microlens (5.1) cameras are given in Tables 6.1 and 6.2, respectively, where $k_{nm}^{(\cdot)}$ denotes the entry (n, m) of the intrinsic matrix and $t_n^{(\cdot)}$ denotes the entry n of the projection center associated with the viewpoint (i, j) or microlens (k, l). Using the values in Table 6.1 and the mappings (6.9) and (6.13), one obtains the characterization of the camera arrays for different values of disparity β.

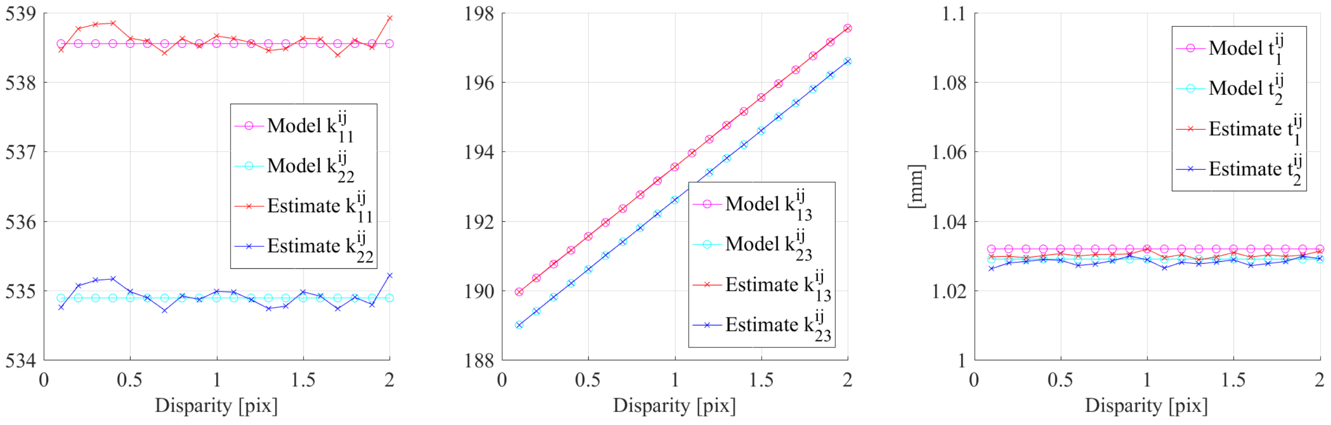
h_{si}	h_s	h_{tj}	h_t	h_{ui}	h_{uk}	h_u	h_{vj}	h_{vl}	h_v
0.0003	-0.0013	0.0003	-0.0013	-0.0011	0.0019	-0.3508	-0.0011	0.0019	-0.3515

Table 6.1: LFIM obtained after calibration of Dataset A [41] with $h_{sk} = h_{tl} = 0$.

k_{11}^{ij}	k_{22}^{ij}	k_{13}^{ij}	k_{23}^{ij}	t_1^{ij}	t_2^{ij}	t_3^{ij}	k_{11}^{kl}	k_{22}^{kl}	k_{13}^{kl}	k_{23}^{kl}	t_1^{kl}	t_2^{kl}	t_3^{kl}
538.6	534.9	189.6	188.6	0.001	0.001	0	881.0	892.0	-307.5	-311.9	-0.081	-0.081	-0.227

Table 6.2: Intrinsic matrices and projection centers for viewpoint and microlens cameras. These values are obtained after applying the mappings in Sections 4.2 and 5.3 with $\Delta i = \Delta j = 1$ and $\Delta k = \Delta l = 1$, respectively.

The characterization of the viewpoint and microlens cameras obtained using the mappings proposed is compared with the characterization obtained by applying (4.1) and (5.1) to the LFIM obtained from the calibration of the sheared versions of the calibration dataset LFs. The sheared LFs are obtained considering different disparities β for the reparameterization of the EPIs (shearing). The disparities considered range from 0.1 to 2.0 pixels. Figure 6.5 depicts the entries of the viewpoint intrinsic matrix and projection center with the disparity β used for shearing considering a unitary displacement from the reference viewpoint (i_r, j_r), *i.e.* $\Delta i = \Delta j = 1$. Similarly, Figure 6.6 depicts the entries of the microlens intrinsic matrix and projection center considering $\Delta k = \Delta l = 1$. Tables 6.3 and 6.4 represent the mean and Standard Deviation (STD) of the errors $\epsilon_{(\cdot)} = \left|(\cdot)^M - (\cdot)^E\right| / \left|(\cdot)^M\right|$, in percentage, for each entry of the intrinsic matrix and projection center for the viewpoint and microlens camera, respectively. In the error $\epsilon_{(\cdot)}$, $(\cdot)^M$ corresponds to the entries obtained from the mappings (6.9) and (6.13), and $(\cdot)^E$ corresponds to the entries obtained from the mappings (4.1) and (5.1).



(a) (b) (c)

Figure 6.5: Variation of viewpoint camera intrinsic matrix and projection center with disparity β for shearing. These entries are estimated considering that $\Delta i = \Delta j = 1$. The scale factors of the intrinsic matrix k^{ij}_{11} and k^{ij}_{22} are represented in **(a)**. The principal point $\left[k^{ij}_{13}, k^{ij}_{23}\right]^T$ is depicted in **(b)**. In **(c)**, the x- and y- components of the projection are presented. The z-component of the projection center is not represented since it is always zero regardless of the disparity β considered for shearing.

k^{ij}_{11}	k^{ij}_{22}	k^{ij}_{13}	k^{ij}_{23}	t^{ij}_{1}	t^{ij}_{2}
0.022 ± 0.018	0.022 ± 0.017	0.004 ± 0.003	0.002 ± 0.002	0.174 ± 0.067	0.100 ± 0.075

Table 6.3: Mean and STD error, in percentage, for each entry of the viewpoint intrinsic matrix and projection center.

k^{kl}_{11}	k^{kl}_{22}	k^{kl}_{13}	k^{kl}_{23}	t^{kl}_{1}	t^{kl}_{2}	t^{kl}_{3}
1.85 ± 6.45	4.45 ± 19.07	1.84 ± 6.45	4.44 ± 19.07	1.89 ± 6.38	1.89 ± 6.38	1.90 ± 6.38
0.41 ± 0.44	0.19 ± 0.18	0.40 ± 0.44	0.18 ± 0.18	0.47 ± 0.42	0.47 ± 0.42	0.48 ± 0.43

Table 6.4: Mean and STD error, in percentage, for each entry of the microlens intrinsic matrix and projection center. First line considers all disparity values while the second line excludes the disparity $\beta = 0.6$.

The viewpoint mapping (6.9) models the changes with the disparity β very accurately (Figure 6.5). In Table 6.3, one can see that the mean error is below 0.2% which shows that the estimate values are in accordance with the mapping (6.9). The difference on the estimated values appears to be the result of the interpolation and discretization that occurs in the shearing operation. This also affects the position of the detected corners that are used in the calibration.

The microlens mapping (6.13) also models the changes with the disparity β very accurately except for $\beta = 0.6$ (Figure 6.6). This disparity value is close to the singularity that occurs for $\beta = -h_{ui}/h_{uk} = 0.611$

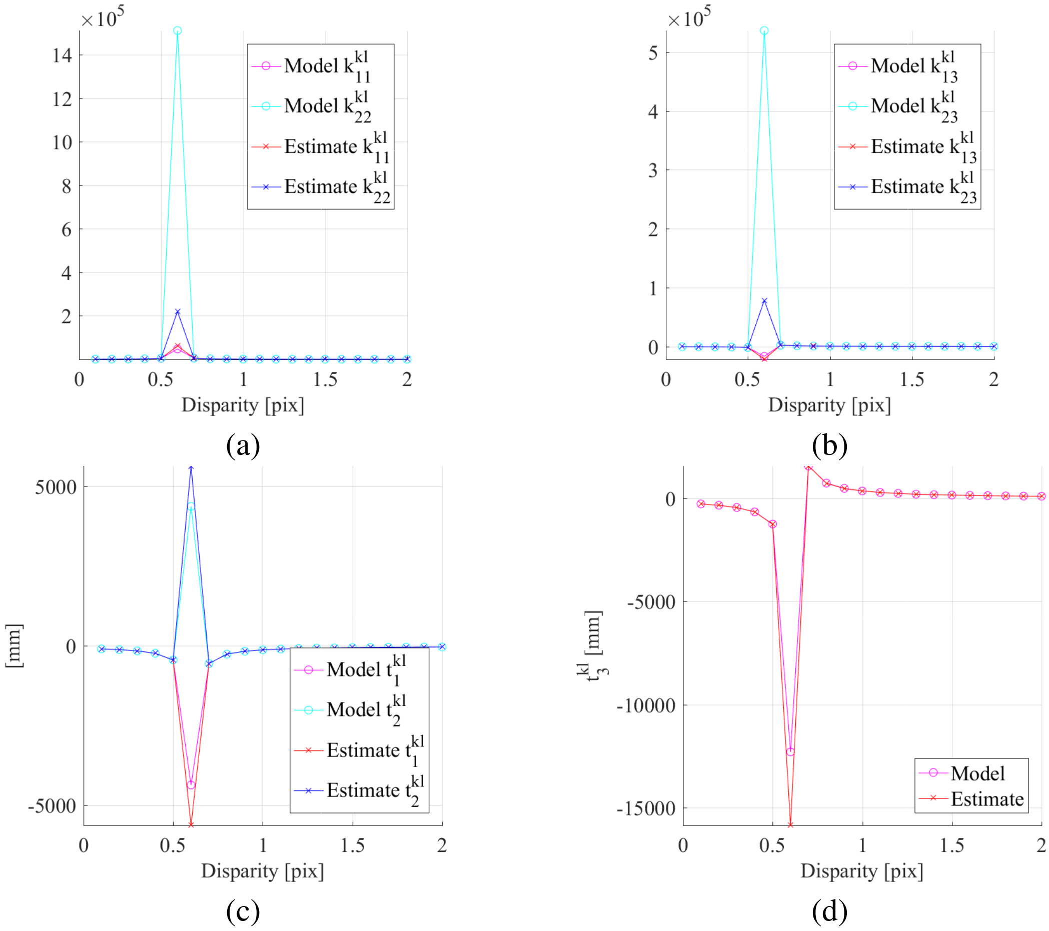


Figure 6.6: Variation of microlens camera intrinsic matrix and projection center with disparity β for shearing. These entries are estimated considering that $\Delta k = \Delta l = 1$. The scale factors of the intrinsic matrix k_{11}^{kl} and k_{22}^{kl} are represented in **(a)**. The principal point $\left[k_{13}^{kl}, k_{23}^{kl}\right]^T$ is depicted in **(b)**. The x- and y- components of the projection are presented in **(c)** while the z-component is presented in **(d)**.

which causes some numerical instability in the mapping. Indeed, in Table 6.4, one can see that the mean error considering all disparity values is below 4.5%. Nonetheless, removing the disparity $\beta = 0.6$, one obtains a mean error below 0.5% which shows that the estimate values are in accordance with the mapping (6.13). Notice that the viewpoint mapping obtains a lower error than the microlens mapping. This can be justified by the strategy of the calibration procedure [41] that calibrates an SPC using detected corners on VIs.

6.2 Plenoptic Camera Array

The microlens or viewpoint camera arrays of lenticular based plenoptic cameras are characterized by having a narrow baseline which limit the reconstruction capabilities of these cameras. Recently, camera arrays of plenoptic cameras started to be used to capture information of the scene [40]. Nonetheless, the strategies to calibrate these cameras consider an independent calibration of each plenoptic camera followed by an estimation of the relative position of each camera. This creates a representation that grows with the number of cameras used in this plenoptic camera array. In this section, one proposes a camera model that does not grow with the cameras considered in the plenoptic camera array and that extends the 5×5 LFIM to represent a coplanar array of plenoptic cameras with the same world focal plane.

6.2.1 Multiple Baseline Camera Array

Let us consider an array composed of equally spaced coplanar plenoptic cameras. Each plenoptic camera is assumed to have the same zoom and focus settings, *i.e.* the same main lens world focal plane.

The LF in the image space acquired by this camera array collects rays $\boldsymbol{\zeta} = [p, g, i, j, k, l]^T$ parameterized by the plenoptic camera (p, g), the pixels (i, j) and the microlenses (k, l) indices. On the other hand, the LF in the object space collects rays that are parameterized by an additional point (q, r) on the parameterization plane Γ. This allows to define a ray $\boldsymbol{\xi} = [q, r, s, t, u, v]^T$ that defines a line whose points are given by

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} q \\ r \\ 0 \end{bmatrix} + \begin{bmatrix} s \\ t \\ 0 \end{bmatrix} + \lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad , \lambda \in \mathbb{R} \quad . \tag{6.20}$$

As seen in Section 4.1, plenoptic cameras can be represented by a viewpoint camera array. The vector $[s, t, 0]^T$ represents the different projection centers enclosed in a plenoptic camera, the viewpoint camera projection centers, while the vector $[q, r, 0]^T$ represents the origin of the coordinate systems of each individual plenoptic camera. The rays in the image space $\boldsymbol{\zeta}$ are mapped to the rays in the object space $\boldsymbol{\xi}$ by a 7×7 matrix $\mathbf{H}_a$:

$$\tilde{\boldsymbol{\xi}} = \mathbf{H}_a \, \tilde{\boldsymbol{\zeta}} \tag{6.21}$$

where

$$\mathbf{H}_a = \begin{bmatrix} \begin{matrix} h_{qp} & 0 \\ 0 & h_{rg} \end{matrix} & \mathbf{0}_{2\times5} \\ \mathbf{0}_{5\times2} & \mathbf{H} \end{bmatrix} \tag{6.22}$$

and $\mathbf{H}$ is the LFIM (4.17).

Viewpoint Camera Array. Let us represent the viewpoint camera array by a parametric projection matrix $\mathbf{P}^{pgij}$ varying with the coordinates (p, g, i, j)

$$\mathbf{P}^{pgij} = \mathbf{K}^{ij} \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{t}^{pgij} \end{bmatrix} {}^{c}\mathbf{T}_w \tag{6.23}$$

where $\mathbf{K}^{ij}$ denotes the intrinsic matrix (4.13) and $\mathbf{t}^{pgij}$ is the projection center of the viewpoint camera (i, j) associated with the plenoptic camera (p, g) defined by

$$\mathbf{t}^{pgij} = \begin{bmatrix} -p \; h_{qp} \\ -g \; h_{rg} \\ 0 \end{bmatrix} + \mathbf{t}^{ij} \tag{6.24}$$

with $[-h_{qp}, -h_{rg}, 0]^T$ denoting the baseline between consecutive plenoptic cameras and $\mathbf{t}^{ij}$ (4.18). Thus, a plenoptic camera array can be considered a multi-baseline plenoptic camera since besides the baseline between plenoptic cameras, one also has the baselines between viewpoint cameras in a plenoptic camera [104].

The model defines one coordinate frame ${}^{c}\mathbf{T}_{w}$ and the same pixel size for all viewpoints while the principal point is different for each viewpoint of a plenoptic camera. Notice that the projections centers are different for each viewpoint on a plenoptic camera and from one plenoptic camera to the other. Notice that $\mathbf{K}^{ij}$ is the same since one assumes identical plenoptic cameras with the same world focal plane.

6.2.2 Corner-based Calibration

The calibration proposed considers the corners of a planar calibration grid of known dimensions as features. In the following, one assumes that the corners in the world coordinate system have been matched with the imaged corners. An imaged corner is defined by a ray $\boldsymbol{\zeta} = [p, g, i, j, k, l]^{T}$ in the image space. The (k, l) coordinates correspond to the pixel coordinates of the detected corners on the VIs while the (i, j) coordinates correspond to the viewpoint coordinates, and (p, g) correspond to the plenoptic camera indices.

Linear Initialization. Considering the mapping in Section 6.2.1, one defines a linear solution for the viewpoint array parameters associated with a multi-baseline plenoptic camera and the extrinsic parameters for each pose of the calibration grid. The linear solution comprises homography, intrinsic and extrinsic parameters estimation steps.

Homography Estimation. Considering the viewpoint projection matrix $\mathbf{P}^{pgij}$ (6.23) with $\mathbf{K}^{ij}$ (4.13) and $\mathbf{t}^{pgij}$ (6.24), a point $\mathbf{m} = [x, y, z]^{T}$ in the object space is projected to a point in the image plane $\mathbf{q} = [k, l]^{T}$ by

$$\tilde{\mathbf{q}} \sim \mathbf{P}^{pgij}\, \tilde{\mathbf{m}} = \mathbf{K}^{ij} \left[{}^{c}\mathbf{R}_{w} \quad {}^{c}\mathbf{t}_{w} + \mathbf{t}^{pgij} \right] \tilde{\mathbf{m}} \tag{6.25}$$

where the symbol $\sim$ denotes equal up to a scale factor. The coplanar grid points allow to define a world coordinate system such that the z-coordinate is zero. In this context, denoting $\tilde{\mathbf{m}} = [x, y, 1]^{T}$, one can redefine the projection (6.25) as $\tilde{\mathbf{q}} \sim \mathbf{H}^{pgij}\, \tilde{\mathbf{m}}$ where

$$\mathbf{H}^{pgij} = \mathbf{K}^{ij}\left[\mathbf{r}_1 \;\; \mathbf{r}_2 \;\; {}^{c}\mathbf{t}_w + \mathbf{t}^{pgij}\right] \tag{6.26}$$

is the parametric homography matrix for the camera (p, g, i, j), and ${}^{c}\mathbf{R}_w = [\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3]$.

The homography matrix $\mathbf{H}^{pgij}$ changes among viewpoints as a result of the principal point shift and baseline defined in Section 6.2.1. Let us consider that $\mathbf{H}^{pgij}$ can be defined from the homography matrix $\mathbf{H}^0$ associated with the viewpoint coordinates $(p, g, i, j) = (0, 0, 0, 0)$, the homography viewpoint change matrix $\mathbf{A}^{ij}$ and the homography plenoptic change matrix $\mathbf{B}^{pg}$ by

$$\mathbf{H}^{pgij} = \underbrace{\begin{bmatrix} h^0_{11} & h^0_{12} & h^0_{13} \\ h^0_{21} & h^0_{22} & h^0_{23} \\ h^0_{31} & h^0_{32} & h^0_{33} \end{bmatrix}}_{\mathbf{H}^0} + \begin{bmatrix} i & 0 & 0 \\ 0 & j & 0 \\ 0 & 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 0 \end{bmatrix}}_{\mathbf{A}^{ij}} + \begin{bmatrix} p & 0 & 0 \\ 0 & g & 0 \\ 0 & 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} 0 & 0 & b_{13} \\ 0 & 0 & b_{23} \\ 0 & 0 & 0 \end{bmatrix}}_{\mathbf{B}^{pg}}. \tag{6.27}$$

Considering the homography projection of a calibration grid corner $\tilde{\mathbf{m}} = [x, y, 1]^T$ in the object space to the image point $\tilde{\mathbf{q}}$ for the camera (p, g, i, j), applying the cross product by $\tilde{\mathbf{q}}$ on each side of the projection equation leads to $[\tilde{\mathbf{q}}]_\times \mathbf{H}^{pgij}\, \tilde{\mathbf{m}} = \mathbf{0}_{3\times1}$, where $\left[(\cdot)\right]_\times$ is a skew-symmetric matrix that applies the cross product. Using the properties of the Kronecker product [93] and solving for each of the unknown parameters, one obtains

$$\left(\tilde{\mathbf{m}}^T \otimes [\tilde{\mathbf{q}}]_\times\right) \mathbf{T} \begin{bmatrix} \mathbf{h}^0 \\ \mathbf{a}^{ij} \\ \mathbf{b}^{pg} \end{bmatrix} = \mathbf{0}_{3\times1} \tag{6.28}$$

where

$$\mathbf{T} = \left[\begin{array}{cccccccccc} & i & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & 0 & j & 0 & 0 & 0 & 0 & 0 & 0 \\ & \multicolumn{8}{c}{\mathbf{0}_{1\times 8}} \\ & 0 & 0 & i & 0 & 0 & 0 & 0 & 0 \\ \mathbf{I}_{9\times 9} & 0 & 0 & 0 & j & 0 & 0 & 0 & 0 \\ & \multicolumn{8}{c}{\mathbf{0}_{1\times 8}} \\ & 0 & 0 & 0 & 0 & i & 0 & p & 0 \\ & 0 & 0 & 0 & 0 & 0 & j & 0 & g \\ & \multicolumn{8}{c}{\mathbf{0}_{1\times 8}} \end{array}\right], \tag{6.29}$$

and $\mathbf{h}^0$, $\mathbf{a}^{ij}$ and $\mathbf{b}^{pg}$ correspond to vectorizations of the matrix $\mathbf{H}^0$, $\mathbf{A}^{ij}$ and $\mathbf{B}^{pg}$ by stacking their columns and removing the zero entries, respectively. The solution $[\mathbf{h}^0, \mathbf{a}^{ij}, \mathbf{b}^{pg}]^T$ for the parametric homography matrix can be estimated using Singular Value Decomposition (SVD).

The parametric homography matrix (6.27) has 17 parameters. According to (6.28), each point correspondence $(\tilde{\mathbf{m}}, \tilde{\mathbf{q}})$ originates three equations with only two being linearly independent. Nonetheless, the restrictions on the viewpoint camera array also originate restrictions on the projections of a point in the object space. Namely, the ray in the image space $\boldsymbol{\zeta}^{pgij} = [p, g, i, j, k, l]^T$ associated with an arbitrary camera (p, g, i, j) can be described from the ray coordinates $\boldsymbol{\zeta}^0 = [0, 0, 0, 0, k_0, l_0]^T$ associated with the camera $(p, g, i, j) = (0, 0, 0, 0)$ by $\boldsymbol{\zeta}^{pgij} = \boldsymbol{\zeta}^0 + [p, g, i, j, i\beta + p\gamma, j\beta + g$ where β corresponds to the disparity of the point defined on the VIs and γ corresponds to the disparity of the point between plenoptic cameras. This reduces the number of linearly independent equations originated by a point in the object space to 6. Thus, one needs at least 3 non-collinear points to obtain the entries of the homography matrix $\mathbf{H}^{pgij}$.

Intrinsic and Extrinsic Estimation. The intrinsic matrix of the viewpoint cameras is the same regardless of the plenoptic camera (p, g) so one can use the method proposed in Section 4.3.1. This method should be changed only if the world focal plane changes among plenoptic cameras.

The extrinsic parameters can be estimated once the intrinsic matrix $\mathbf{K}^{ij}$ is known. From (6.26), the rotation matrix ${}^{c}\mathbf{R}_{w} = [\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3]$ is recovered considering

$$\mathbf{r}_1 = \lambda {\mathbf{K}^{ij}}^{-1}\mathbf{h}_1 \ , \ \mathbf{r}_2 = \lambda {\mathbf{K}^{ij}}^{-1}\mathbf{h}_2 \ , \text{ and } \mathbf{r}_3 = \mathbf{r}_1 \times \mathbf{r}_2 \tag{6.30}$$

with $\lambda = 1/\left\|{\mathbf{K}^{ij}}^{-1}\mathbf{h}_1\right\| = 1/\left\|{\mathbf{K}^{ij}}^{-1}\mathbf{h}_2\right\|$. The translation ${}^{c}\mathbf{t}_{w}$ and projection center $\mathbf{t}^{pgij}$ are recovered solving the following system of equations

$$\lambda \mathbf{h}_3 = \begin{bmatrix} \mathbf{K}^{ij} & -p\mathbf{k}_1 & -g\mathbf{k}_2 & -i\mathbf{k}_1 & -j\mathbf{k}_2 \end{bmatrix} \begin{bmatrix} {}^{c}\mathbf{t}_{w} \\ h_{qp} \\ h_{rg} \\ h_{si} \\ h_{tj} \end{bmatrix} \tag{6.31}$$

where $\mathbf{k}_m$ corresponds to the m-th column of the parametric intrinsic matrix $\mathbf{K}^{ij}$.

Nonlinear Optimization. The linear solution is refined and radial distortion [25] is considered on the coordinates (u, v). Namely, the undistorted rays in the object space $\boldsymbol{\xi}^{u} = [q, r, s, t, u^{u}, v^{u}]^{T}$ are defined from distorted rays in the object space $\boldsymbol{\xi} = [q, r, s, t, u, v]^{T}$ by (4.34) that is described by the distortion vector $\mathbf{d} = (k_1, k_2, k_3, b_u, b_v)$. In the nonlinear optimization, one minimizes the ray re-reprojection error. This optimization refines the intrinsic parameters $\mathbf{H}_a$, the extrinsic parameters $\mathbf{R}_p$ (parameterized by Rodrigues formula [48]) and $\mathbf{t}_p$, $p = 1, \ldots, P$ where P is the number of poses, and the distortion vector $\mathbf{d}$:

$$\underset{\mathbf{H}_a, \mathbf{R}_p, \mathbf{t}_p, \mathbf{d}}{\arg\min} \sum_{p=1}^{P} \sum_{n=1}^{N_p} \sum_{c=1}^{C} \Lambda \left(\eta_{n}^{c} \left(\mathbf{H}_a, \mathbf{d} \right), \mathbf{R}_p \, \mathbf{m}_n + \mathbf{t}_p \right) \tag{6.32}$$

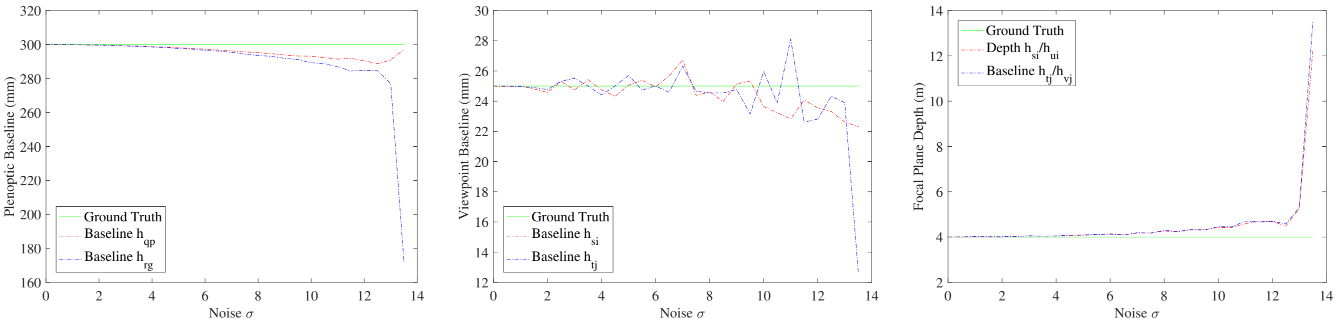
where N_p corresponds to the number of corners detected on a pose p, C corresponds to the number of viewpoint cameras, $\Lambda(\cdot)$ defines the point-to-ray distance [41], η_n^c defines the undistorted ray coordinates $\boldsymbol{\xi}^u$ after mapping the ray in the image space $\boldsymbol{\zeta}_n^c$ associated with the viewpoint camera c and corner n to the ray in object space (6.21) and followed by distortion rectification (4.34). $\mathbf{m}_n$ defines the 3D corner point in the world coordinate system. The nonlinear optimization is solved using the trust-region-reflective algorithm [35], where a sparsity pattern for the Jacobian matrix is provided. The number of parameters over which one optimizes is 10 for the intrinsic parameters, 5 for the lens distortion parameters, and $6P$ for the extrinsic parameters.

6.2.3 Experimental Results

The calibration proposed for a plenoptic camera array is evaluated on a synthetic dataset. The synthetic dataset is obtained extending the 4D LF Benchmark add-on for Blender [68]. This tool simulates a plenoptic camera with rectangular sampling by considering a set of coplanar cameras. The multi-baseline plenoptic camera is simulated by placing a set of identical plenoptic cameras equally spaced in a plane.

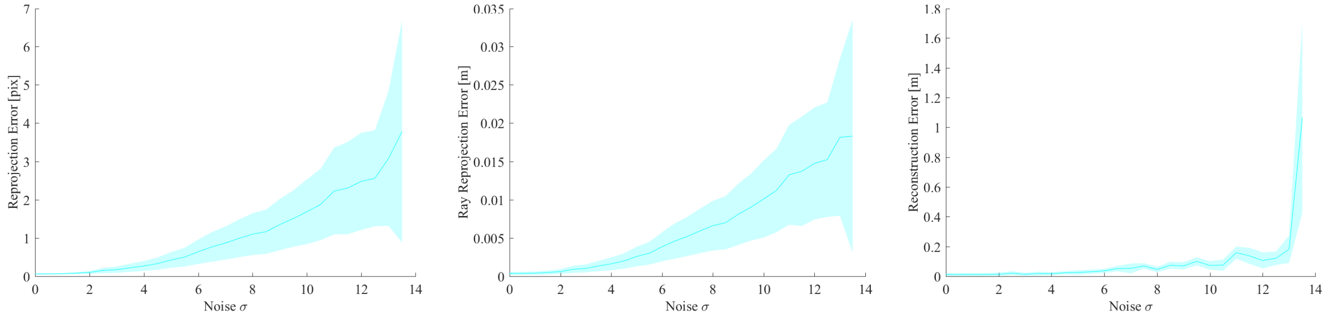
The multi-baseline plenoptic camera calibration dataset is acquired assuming an array of 3×3 plenoptic cameras and comprise 12 calibration poses. The plenoptic cameras are spaced by 0.3 m and are focused at 4.0 m. Each plenoptic camera is composed of 5×5 viewpoint cameras spaced by 25 mm.

The calibration is performed using the corner points detected using the feature detector [78] and compared with the results of the calibration using the ground truth corner points. The calibration using the detected points is repeated several times with different levels of noise added to the location of the corner points. The noise is assumed to be Gaussian noise with zero mean and increasing variance. The results are summarized in Figure 6.7. The reprojection error as well as the ray reprojection and reconstruction errors are provided in Figure 6.8.



(a) (b) (c)

Figure 6.7: Multi-baseline plenoptic camera parameters estimation with added Gaussian noise. The plenoptic array baseline is depicted in **(a)** while the viewpoint baseline is depicted in **(b)**. In **(c)** the focus depth is represented. The ground truth values are depicted in red.



(a) (b) (c)

Figure 6.8: RMS of reprojection **(a)**, ray reprojection **(b)** and reconstruction errors **(c)** with added Gaussian noise.

Figure 6.7 shows that the calibration method proposed allows to recover the parameters of the plenoptic and viewpoint cameras even when the location of the corners is highly affected by noise. More specifically, the baselines and the focus depth have errors above 10% for $\sigma \geq 10.0$. In addition, in Figure 6.8, one shows that the reprojection error is sub-pixel for $\sigma \leq 7.5$ and the reconstruction error obtained is below 100 mm for $\sigma \leq 10.5$ which shows the accuracy of the calibration method and the camera model proposed.

6.3 Chapter Summary

The rays captured by a plenoptic camera allow to define multiple camera arrays besides the viewpoint and microlens camera arrays [21, 104, 108] described in Sections 4.1 and 5.1. In this chapter, is shown that by selecting different combination of rays, one can generate new views of the scene at different depths. Namely, resampling the LF according with the constraint (6.1) one defines new camera arrays whose geometry is defined in Section 6.1.1. The resulting EPI geometry shows that the world focal plane is dependent on the resampling β.

In Section 6.2.1, was extended the LFIM to represent a coplanar array of plenoptic cameras with the same world focal plane. This representation allows to describe a multi-baseline camera array: the baseline among the viewpoint cameras within a plenoptic camera and the baseline among plenoptic cameras. Using the viewpoint camera array model, one proposed a calibration procedure for the plenoptic camera array (Section 6.2.2).

In the next chapter, the scope of the concepts presented until now will change to focus on a natural application of the LF, namely, depth estimation.

Chapter 7

Reconstruction

In a plenoptic camera, a point in the object space is projected into multiple points in the image sensor which allow to recover disparity and depth. In this chapter, are described several reconstruction approaches ranging from sparse to dense reconstruction. These reconstruction approaches are improved considering the projection model of the plenoptic cameras described in the previous chapters and the original construct of affine Lightfield (LF). In addition, an efficient dense reconstruction methodology is proposed that allows to obtain disparity estimates for the full 4D LF.

7.1 Related Work

The multiple projections of a point in a plenoptic camera allow to recover disparity and depth assuming no particular position for the cameras, *e.g.* using multiview stereo [3], or assuming the cameras define a linear path, *e.g.* using the Epipolar Plane Image (EPI) geometry [46]. The LF obtained by a plenoptic camera is equivalent to the one obtained by a camera array whose cameras are regularly arranged and spaced (Chapters 4 and 5). Thus, a 3D point on a Lambertian surface defines a plane of constant intensity in the LF whose orientation represents the depth of the point [39].

The plane's orientation can be estimated considering gradient based approaches using standard image gradient operators [39, 89] or structure tensors [139] due to the very narrow baseline. Nonetheless, these approaches limit the disparity range that can be estimated accurately to one pixel [46]. Shearing of the LF [46] increase the disparity range while maintaining the gradient operators size constant.

Another strategy to estimate the plane's orientation consists on testing a predefined disparity hypothesis by shearing the LF and evaluating correspondence, defocus and shading cues on the resulting LF [134, 136]. Recently, the concept of Surface Camera Images (SCams) [29] has been introduced to identify types of surfaces (Lambertian or specular) and occlusions that allow to adapt the metrics used to evaluate correspondences. Nonetheless, these methodologies only allow to obtain sparse disparity estimates.

The sparse disparity estimates can be used to obtain dense disparity maps using variational approaches. Wanner *et al.* [56, 141] proposed a global optimization framework that is based on the regularization and integration of the disparity maps obtained from the EPIs. This framework can be preceded of a labeling scheme to impose visibility constraints that imply a discretization of the disparity values. This step is computationally expensive and the discretization reduces the accuracy of the disparity estimation. Hence, Wanner *et al.* [139] considered a more efficient approach by performing a fast denoising of the initial disparity estimates that result from selecting the disparities obtained from a small subset of the LF. This approach allows to retrieve a dense disparity map only for the central viewpoint camera. Thus, in this chapter, is formalized a data fusion problem with total variation regularization using the Alternating Direction Method of Multipliers (ADMM) to recover a dense disparity map for the full LF.

In recent years, deep neural networks have also appeared to retrieve disparity from LF [65, 129]. These networks use convolutional neural networks to more precisely estimate disparity from the LF intrinsic cues and performing propagation of these estimates to regions where these cues are absent. More specifically, Shin *et al.* [129] recovers disparity using EPI geometry cues while Hazirbas *et al.* [65] presented the Deep Depth From Focus Network (DDFFNet) which uses focus cues (focal stack).

7.2 Point Reconstruction

In the reconstruction problem, one wants to determine the point in the object space whose rays where projected into specific points of the LF in the image space. The LF obtained by a plenoptic camera is equivalent to the one obtained by a camera array so one strategy is to use a multiview stereo reconstruction approach.

Let us consider that one has a set of L rays in the image space that correspond to a given point $\mathbf{m} = [x, y, z]^T$ in the object space and that the Lightfield Intrinsic Matrix (LFIM) $\mathbf{H}$ is known. This allows to convert the set of rays in the image space, $\mathbf{\Phi}_1, \ldots, \mathbf{\Phi}_L$, to a set of rays in the object space $\mathbf{\Psi}_1, \ldots, \mathbf{\Psi}_L$. Using the relationship between a point $\mathbf{m}$ and the rays in the object space (3.10), one obtains for the n-th ray $x - z\, u_n = s_n$ and $y - z\, v_n = t_n$ which in matrix form corresponds to

$$\begin{bmatrix} 1 & 0 & -u_n \\ 0 & 1 & -v_n \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} s_n \\ t_n \end{bmatrix} \quad . \tag{7.1}$$

From equation (7.1), for each ray $\mathbf{\Psi}_n$ one obtains a set of two equations. The reconstruction problem has three unknowns to determine, hence, one needs at least two point-ray correspondences to determine the corresponding point $\mathbf{m}$.

Generalizing the equation (7.1) for the rays $\mathbf{\Psi}_1, \ldots, \mathbf{\Psi}_L$, and replacing those rays by the rays in the image space $\mathbf{\Phi}_1, \ldots, \mathbf{\Phi}_L$, one has

$$\begin{bmatrix} 1 & 0 & -\mathbf{h}_3\,\boldsymbol{\Phi}_1 \\ 0 & 1 & -\mathbf{h}_4\,\boldsymbol{\Phi}_1 \\ \vdots & \vdots & \vdots \\ 1 & 0 & -\mathbf{h}_3\,\boldsymbol{\Phi}_L \\ 0 & 1 & -\mathbf{h}_4\,\boldsymbol{\Phi}_L \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \mathbf{h}_1\,\boldsymbol{\Phi}_1 \\ \mathbf{h}_2\,\boldsymbol{\Phi}_1 \\ \vdots \\ \mathbf{h}_1\,\boldsymbol{\Phi}_L \\ \mathbf{h}_2\,\boldsymbol{\Phi}_L \end{bmatrix} \tag{7.2}$$

where $\mathbf{h}_n$ corresponds to the n-th row of the LFIM $\mathbf{H}$. This is a problem that can be readily solved using a least-squares method.

7.2.1 Imposing Projection Geometry Cues

The multiview stereo reconstruction methodology does not impose any prior knowledge on the rays in the image space that originate at a given 3D point. Namely, this reconstruction methodology does not consider that the cameras in the camera array are regularly arranged and spaced. Hence, the reconstruction is as good as the precision of the detections, maintaining all parameters of the optical system constant.

The rays in the image space, due to the discretization that occurs at the image sensor, do not define a line in the ray-space defined by the pair of coordinates (i, k) and (j, l) but a staircase (Figure 3.3). Therefore, the precision of the detections and, consequently, the reconstruction is likely to improve if one imposes the rays $\boldsymbol{\Phi}_n$ in the ray-spaces to define a line. Let us call these lines in the ray-spaces as the projection geometry cues.

Let us incorporate the projection cues as a prior knowledge on the rays in the image space $\boldsymbol{\Phi}_1, \ldots, \boldsymbol{\Phi}_L$. This can be achieved by considering the point reconstruction from the lines in each of the ray-spaces (i, k) and (j, l) instead of using the point-ray correspondences directly. Namely, rewriting the projection equation (3.14) as

$$\begin{cases} \underbrace{(h_{si} + z\, h_{ui})}_{a_1}\, i + \underbrace{(h_{sk} + z\, h_{uk})}_{b_1}\, k + \underbrace{h_s + z\, h_u - x}_{c_1} = 0 \\ \underbrace{(h_{tj} + z\, h_{vj})}_{a_2}\, j + \underbrace{(h_{tl} + z\, h_{vl})}_{b_2}\, l + \underbrace{h_t + z\, h_v - y}_{c_2} = 0 \end{cases}, \tag{7.3}$$

one defines the relationship between the point in the object space and the line parameters $\theta_{ik} = [a_1, b_1, c_1]^T$ and $\theta_{jl} = [a_2, b_2, c_2]^T$ that define the lines in the ray-spaces (i, k) and (j, l), respectively. From these equations, one can see that, for a given point, the line parameters are fixed while the coordinates of the LF in the image space may vary. The line parameters are obtained by fitting lines to the collection of coordinate pairs (i, k) and (j, l) of the rays $\mathbf{\Phi}_1, \ldots, \mathbf{\Phi}_L$ in the respective ray-space. Let us define the arrays $\mathbf{\Phi}_n^{ik} = [i_n, k_n, 1]^T$ and $\mathbf{\Phi}_n^{jl} = [j_n, l_n, 1]^T$ containing the coordinates (i, k) and (j, l) of the n-th correspondence. The line parameters can be estimated using a least-squares minimization using the L point-ray correspondences

$$\arg\min_{\theta_{(\cdot)}} \sum_{n=1}^{L} \left| \theta_{(\cdot)}^T\, \mathbf{\Phi}_n^{(\cdot)} \right|^2 \quad \text{s.t.} \quad \left\| \theta_{(\cdot)} \right\|^2 = 1 \tag{7.4}$$

where $(\cdot)$ represents either of the pair of coordinates (i, k) and (j, l) according to the ray-space that is being analyzed. These estimates for the line parameters θ_{ik} and θ_{jl} can then be used to estimate the point $\mathbf{m}$

$$\begin{bmatrix} 0 & 0 & h_{ui} & -a_1 & 0 \\ 0 & 0 & h_{vj} & 0 & -a_2 \\ 0 & 0 & h_{uk} & -b_1 & 0 \\ 0 & 0 & h_{vl} & 0 & -b_2 \\ -1 & 0 & h_u & -c_1 & 0 \\ 0 & -1 & h_v & 0 & -c_2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ \lambda_{ik} \\ \lambda_{jl} \end{bmatrix} = - \begin{bmatrix} h_{si} \\ h_{tj} \\ h_{sk} \\ h_{tl} \\ h_s \\ h_t \end{bmatrix} . \tag{7.5}$$

Remember that the line parameters are defined up to a scale factor, therefore, the scale factors λ_{ik} and λ_{jl} associated with each fitting should also be estimated to recover the correct coordinates for the point $\mathbf{m}$.

The reconstruction methodology proposed has 5 unknowns and 6 equations which allows to obtain a solution for the point in the object space using a least squares method, for example. On the other hand, for the estimation of the line parameters, due to the constraint in (7.4), one needs at least two point-ray correspondences to determine the three unknowns in each of the ray-spaces. A given correspondence contributes only with one equation for each of the ray-spaces. Notice that the optimization can be simplified by dividing (7.3) by b_1 and b_2, respectively. This assumes that the singularity z_s^2 described in Section 3.4.2 will not occur. In fact, for most of the experiments performed, this singularity occurs for points behind the camera. The reconstruction methodology imposing the projection geometry cues is detailed in Algorithm 2, where one considers $\mathbf{\Phi}^{(\cdot)}$ as the collection of coordinates $(\cdot)$ of the rays in the image space.

```
Algorithm 2: Reconstruct scene point $\mathbf{m}$
Input  : Projection Rays: $\{\mathbf{\Phi}_1, \dots, \mathbf{\Phi}_L\}$
         Parameters: $\mathbf{H}$
Output: Scene point: $\mathbf{m} = [x, y, z]^T$
1 Obtain $\boldsymbol{\theta}_{ik}$ by fitting a line to $\left(\mathbf{\Phi}^i,\ \mathbf{\Phi}^k\right)$ using equation (7.4) ;
2 Obtain $\boldsymbol{\theta}_{jl}$ by fitting a line to $\left(\mathbf{\Phi}^j,\ \mathbf{\Phi}^l\right)$ using equation (7.4) ;
3 Reconstruct $\mathbf{m}$ using equation (7.5)
```

7.2.2 Experimental Results

In this section, are compared the two methods described in Section 7.2 by performing point reconstruction for points in the object space at different depths. Hence, let us consider the LFIM $\mathbf{H}$ provided as a result of the calibration of Dataset B [41]. The entries obtained are presented in Table 7.1.

In this experiment, the accuracy at each depth is evaluated by randomly selecting $P = 500$ points from the Field of View (FOV) of the plenoptic camera and computing the reconstruction error after projec-

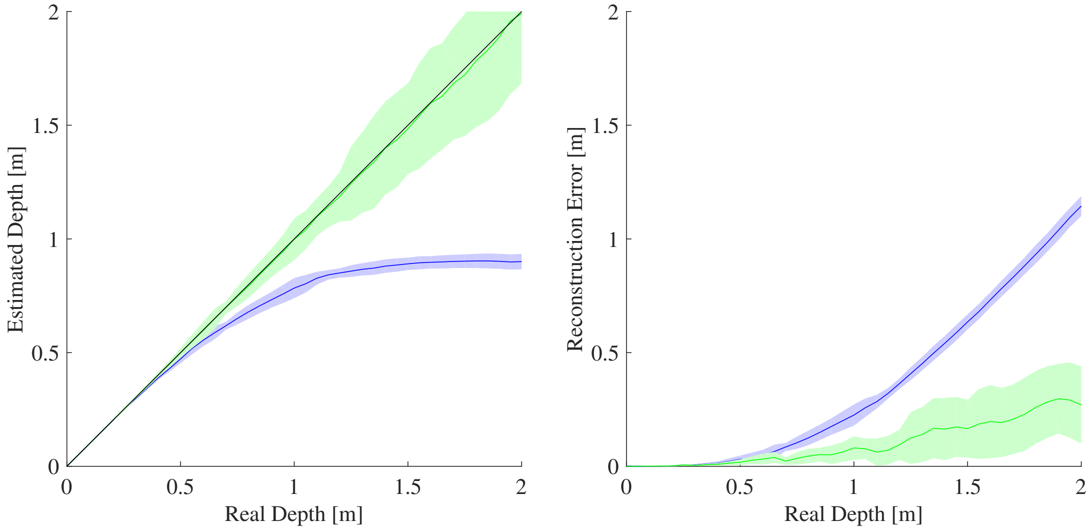
h_{si}	h_{sk}	h_s	h_{tj}	h_{tl}	h_t
4.0003e-04	-9.3810e-05	1.5871e-02	3.9680e-04	-9.3704e-05	1.5867e-02

h_{ui}	h_{uk}	h_u	h_{vj}	h_{vl}	h_v
-1.5833e-03	1.9043e-03	-3.4762e-01	-1.5551e-03	1.9014e-03	-3.3817e-01

Table 7.1: LFIM entries considered for evaluating the point reconstruction methods.

tion and reconstruction using the two methods described. In the projection step, one considers two different sources for the projection error: (i) rounding the pixels (i, j) and the microlenses (k, l) to the nearest integer and (ii) adding noise that follows a Gaussian distribution with zero mean and different Standard Deviations (STDs). The reconstruction error is defined as the distance between the reconstructed point $\hat{\mathbf{m}}_i$ and the generated point $\mathbf{m}_i$ in the object space. The mean reconstruction error r_e is defined by (4.37). The depth values evaluated ranged between 0.01 and 2.00 m. The reconstruction error and the estimated depth of these simulations are provided in Figures 7.1-7.4.

In Figure 7.1, one depicts the reconstruction error when the coordinates of the rays in the image space are affected by a rounding error. The point reconstruction using directly the rays in the image space (blue region) start to deviate from the ground truth at 0.65 m while the reconstruction imposing the projection geometry cues (green region) start to deviate from the ground truth at 1.30 m. The deviation is assumed to occur when the mean reconstruction error r_e normalized by the ground truth depth is greater than 10%. Figure 7.1.a shows that the mean value for the depth estimates using the projection cues are in accordance with the ground truth for the entire depth range tested. Namely, the maximum deviation from the ground truth normalized by the ground truth depth is 15.0% which is significantly lower than the 55.0% obtained for the point reconstruction applied directly to the rays $\mathbf{\Phi}_i$. Nonetheless, the STD normalized by the ground truth depth increases significantly at 1.20 m which makes the depth estimates less reliable. Additionally, one can see that the error on the (x, y) coordinates increase more rapidly than the error on the z-coordinate with the real depth of a point.



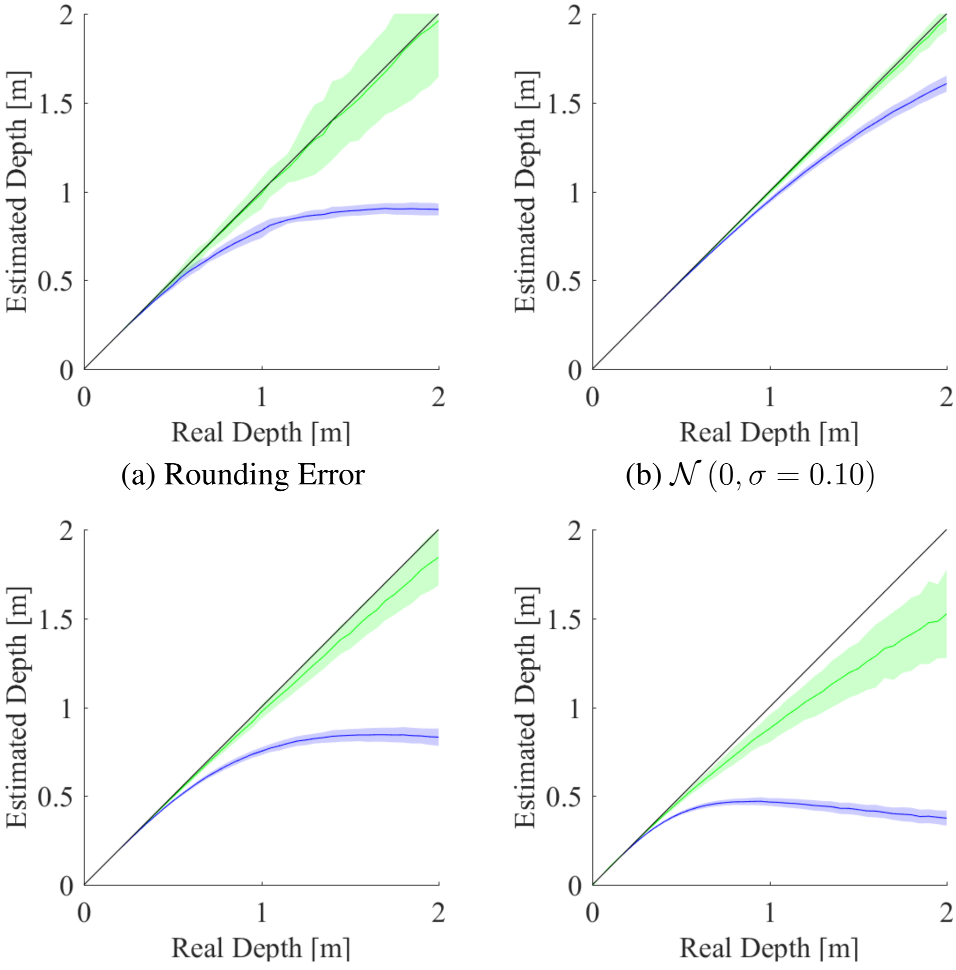
(a) Reconstructed Depth

(b) Reconstruction Error

Figure 7.1: Results from reconstructing randomly generated points at depths ranging from 0.01 to 2.00 m. The projection error is modeled as a rounding of the coordinates of the rays in the image space to the nearest integer. The reconstructed depth is depicted in **(a)** while the reconstruction error using the (x, y, z) coordinates is depicted in **(b)**. The blue region corresponds to the point reconstruction (7.2) and the green region corresponds to the point reconstruction imposing projection geometry cues (7.5).

Figure 7.2 depicts the results assuming that the projection error corresponds to Gaussian noise and that it affects all coordinates of the rays in the image space. In this figure, one can see that the point reconstruction imposing projection geometry cues (7.5) provides better results than the point reconstruction using directly the rays in the image space (7.2) independently of the noise content and source (rounding or Gaussian).

Additionally, since point reconstruction methodologies consider features detected with sub-pixel precision on images obtained from the LF, one models the error introduced by the feature detectors as a Gaussian distribution with zero mean and different STDs. The LF allows to obtain Viewpoint Images (VIs) and Microlens Images (MIs) by fixing either the (i, j) or (k, l) coordinates, respectively. Hence, in Figure 7.3, one considers that the feature detectors introduce error only in the (k, l) coordinates while the coordinates (i, j) are rounded to the nearest integer, *i.e.* features are detected on VIs. In Figure 7.4, one considers that the feature detectors introduce error only in the (i, j) coordinates while the coordi-



(a) Rounding Error

(b) $\mathcal{N}(0, \sigma = 0.10)$

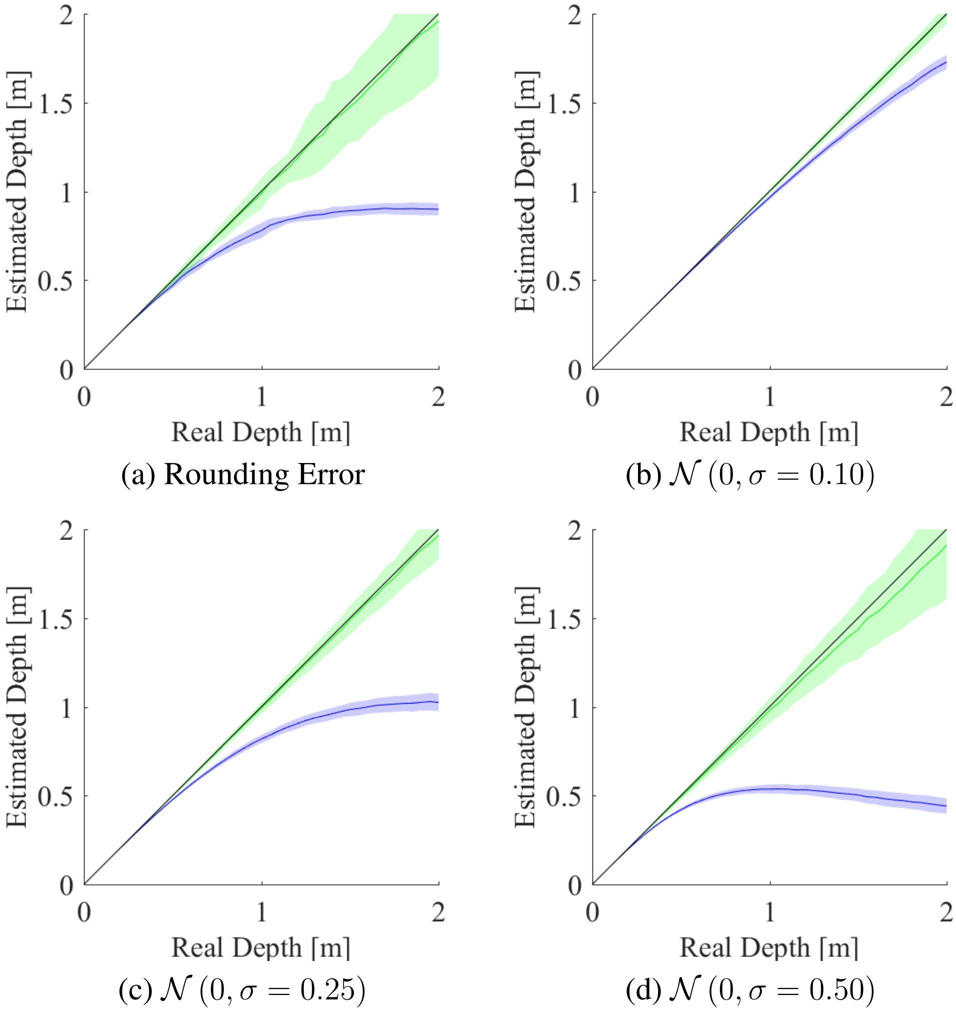
(c) $\mathcal{N}(0, \sigma = 0.25)$

(d) $\mathcal{N}(0, \sigma = 0.50)$

Figure 7.2: Reconstructed depth for randomly generated points at depths ranging from 0.01 to 2.00 m. The projection error modeled as additive Gaussian noise affects all coordinates of the rays in the image space **(b-d)**. The point reconstruction applied to the projection rays $\mathbf{\Phi}_n$ (7.2) is presented in blue while the point reconstruction from line parameters (7.5) is presented in green. The mean for the estimated depth is presented as a darker line and the brighter shaded areas correspond to the STD. The depth ground truth is represented with a black line.

nates (k, l) are rounded to the nearest integer, *i.e.* features are detected on MIs like in [21]. These figures continue to show that the point reconstruction imposing projection geometry cues (7.5) gives better results. Furthermore, one can see that the variance of the reconstructed depth is greater when adding noise to the coordinates (i, j). This indicates that the reconstruction is more robust for noise added to the coordinates (k, l).

As suggested, imposing the projection geometry cues allows to im-



(a) Rounding Error

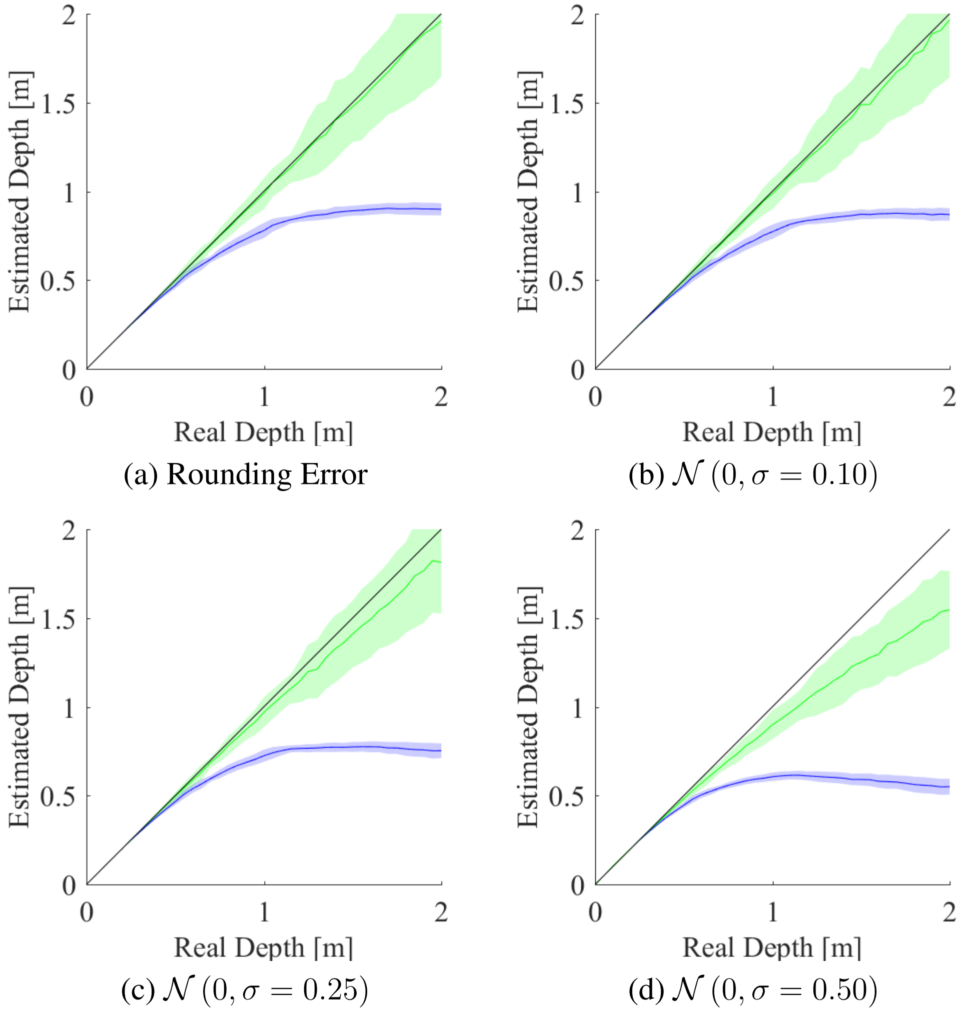
(b) $\mathcal{N}\left(0, \sigma = 0.10\right)$

(c) $\mathcal{N}\left(0, \sigma = 0.25\right)$

(d) $\mathcal{N}\left(0, \sigma = 0.50\right)$

Figure 7.3: **VI Feature Detector Case**. Reconstructed depth for randomly generated points at depths ranging from 0.01 to 2.00 m. The error modeled as additive Gaussian noise affects coordinates (k, l) of the rays in the image space **(b-d)**. The point reconstruction applied to the projection rays $\mathbf{\Phi}_n$ (7.2) is presented in blue while the point reconstruction from line parameters (7.5) is presented in green. The mean for the estimated depth is presented as a darker line and the brighter shaded areas correspond to the STD. The depth ground truth is represented with a black line.

prove the depth reconstruction. More specifically, assuming the pixels (i, j) and the microlenses (k, l) are integers, the reconstruction using line parameters allows the ray coordinates to be real. Let us consider the depth error ε_z for a binocular stereo configuration $\varepsilon_z = \frac{z^2}{b\,f}\varepsilon_d$, where b is the baseline length, f is the focal length, z is the depth of a given point in the object space, and ε_d corresponds to the disparity error. For a given depth of a point, maintaining all parameters constant, the depth error can only decrease by reducing the disparity error. This can be achieved



(a) Rounding Error (b) $\mathcal{N}(0, \sigma = 0.10)$ (c) $\mathcal{N}(0, \sigma = 0.25)$ (d) $\mathcal{N}(0, \sigma = 0.50)$

Figure 7.4: **MI Feature Detector Case**. Reconstructed depth for randomly generated points at depths ranging from 0.01 to 2.00 m. The error modeled as additive Gaussian noise affects coordinates (i, j) of the rays in the image space **(b-d)**. The point reconstruction applied to the projection rays $\mathbf{\Phi}_n$ (7.2) is presented in blue while the point reconstruction from line parameters (7.5) is presented in green. The mean for the estimated depth is presented as a darker line and the brighter shaded areas correspond to the STD. The depth ground truth is represented with a black line.

by increasing the precision of the detections, which is achieved with the reconstruction imposing the projection geometry cues.

7.3 Gradient-based Reconstruction

The LF conveys information that allows to estimate the depth of the objects in the scene and the narrow baseline between the viewpoint and microlens cameras of a plenoptic camera allows to use gradient operators to estimate depth.

Let us denote the 4D LF as $L(i, j, k, l)$ which maps an intensity value to the light ray whose direction is defined by the intersection with the viewpoint camera plane Π at (i, j) and the image plane Γ at (k, l):

$$L : (i, j, k, l) \in \mathbb{R}^4 \mapsto I \in \mathbb{R}^{N_c} \tag{7.6}$$

where N_c is the number of channels. For example, a scalar valued LF has $N_c = 1$ channels and a vector valued LF has $N_c > 1$ channels. As mentioned in Section 2.5.2, an EPI can be obtained readily from the LF considering 2D slices. More specifically, the EPI can be obtained by fixing a pair of coordinates (i_n, k_m) or (j_n, l_m) (Figure 7.5), and considering a (j, l) or (i, k) slice of the LF, respectively

$$E_{j_n,l_m}(i, k) = L(i, j_n, k, l_m) \tag{7.7}$$

where $\mathbf{E}_{j_n,l_m}$ denotes the EPI fixing the pair of coordinates (j_n, l_m). To simplify the notation, the subscript parameters (j_n, l_m) will not be included in the following expressions. In the EPI, a point in space is projected onto a line [22] whose slope is related to its depth by

$$\frac{\Delta k}{\Delta i} = k_u \frac{\Delta x_0}{z} + \Delta u_0 \tag{7.8}$$

where $\frac{\Delta k}{\Delta i}$ corresponds to the disparity, k_u to the scale factor, Δx_0 to the baseline between the viewpoint cameras, z to the depth of a point $\mathbf{m}$ in the object space, and Δu_0 to the principal point shift as explained in Section 4.1.3. Depending on the direction of the movement considered to create the EPIs, (7.8) can be affected by a minus sign.

7.3.1 EPI Disparity Estimation

The most common approach for computing slopes in the EPIs is to use the gradient or the structure tensor [46, 139].

Gradient Estimation. The estimation of disparity using gradients was introduced by Dansereau *et al.* [39]. In this work, it is established the relationship between the ratio of the gradients and the slope $\frac{\Delta i}{\Delta k}$ of the

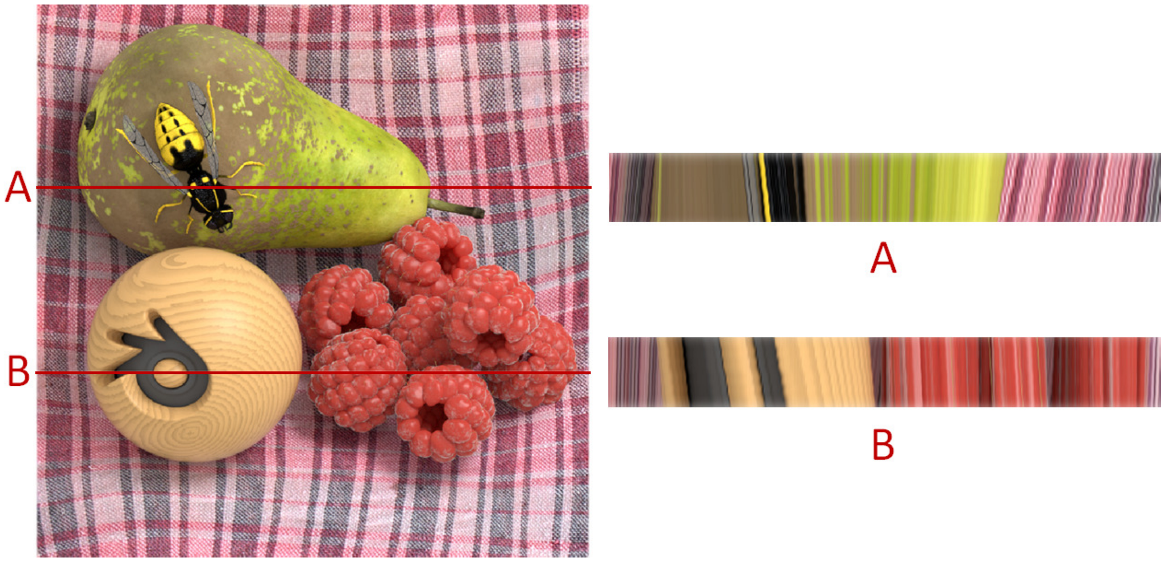


Figure 7.5: Representation of the spatiotemporal characteristics of the EPI. On the left, a static scene from the dataset *Still* [140] is displayed. This image identifies two horizontal slices that are used to obtain the EPIs. On the right the EPIs obtained from slicing and stacking the sequence of VIs at position A and B are depicted.

lines in the EPI. Namely, the main gradient direction is orthogonal to the projected line in the EPI (Figure 7.6)

$$\frac{\Delta i}{\Delta k}\left(\mathbf{n}\right) = -\frac{1}{m_{\nabla}\left(\mathbf{n}\right)} \tag{7.9}$$

where $\mathbf{n} = \left[i, k\right]^T$. The main gradient direction can be determined from the EPI gradients by $m_{\nabla}\left(\mathbf{n}\right) = \frac{E_i(\mathbf{n})}{E_k(\mathbf{n})}$. Thus,

$$\frac{\Delta i}{\Delta k}\left(\mathbf{n}\right) = -\frac{E_k\left(\mathbf{n}\right)}{E_i\left(\mathbf{n}\right)} \tag{7.10}$$

where $\mathbf{E}_i = \nabla_i \mathbf{E}$ and $\mathbf{E}_k = \nabla_k \mathbf{E}$ are the image gradients in the i- and k-direction. Assuming a plenoptic camera defined by the LFIM (4.17), the depth of a point is defined by

$$z_{ik}\left(\mathbf{\Phi}\right) = -\frac{h_{si}}{h_{ui} - m_{\nabla}\left(\mathbf{\Phi}\right) h_{uk}} \quad . \tag{7.11}$$

The relationship between disparity and the LF gradients can be derived from the optical flow [69]. As for the EPI, the temporal dimen-

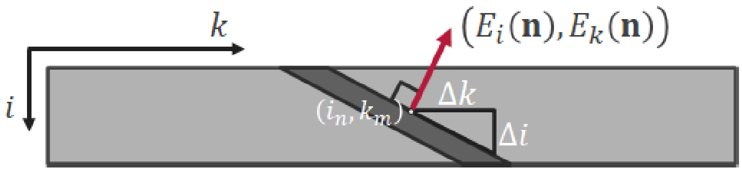


Figure 7.6: Relationship between the EPI gradients and the projection line of a point in the object space.

sion in the LF can be simulated by considering an arbitrary path $\mathbf{p}(t) = \big(i(t), j(t)\big)$ through the several viewpoints. Hence, the brightness constancy constraint between point-ray correspondences is defined as

$$L\big(\mathbf{\Phi}(t+\Delta t)\big) = L\big(\mathbf{\Phi}(t)\big) \tag{7.12}$$

where $\mathbf{\Phi}(t) = \big[i(t), j(t), k, l\big]^T$. Rewriting $\mathbf{\Phi}(t+\Delta t) = \mathbf{\Phi}(t) + \Delta\mathbf{\Phi}(\Delta t)$, and assuming that the displacement from one viewpoint to another and the displacement of the pixel positions of a 3D point on these viewpoints is small, one can approximate the left term using a Taylor series expansion

$$L\big(\mathbf{\Phi}(t) + \Delta\mathbf{\Phi}(\Delta t)\big) \approx L\big(\mathbf{\Phi}(t)\big) + \nabla\mathbf{L}\big(\mathbf{\Phi}(t)\big)^T \Delta\mathbf{\Phi}(\Delta t) \tag{7.13}$$

where $\Delta\mathbf{\Phi}(\Delta t) = \left[\frac{di}{dt}\Delta t, \frac{dj}{dt}\Delta t, \Delta k, \Delta l\right]^T$. This allows to define the brightness constancy constraint as

$$\nabla\mathbf{L}\big(\mathbf{\Phi}(t)\big)^T \Delta\mathbf{\Phi}(\Delta t) = 0 \tag{7.14}$$

where $\nabla\mathbf{L}(\mathbf{\Phi}) = \big[L_i(\mathbf{\Phi}), L_j(\mathbf{\Phi}), L_k(\mathbf{\Phi}), L_l(\mathbf{\Phi})\big]^T$ is the LF gradient vector at point $\mathbf{\Phi}$ with $\mathbf{L}_{(\cdot)} = \nabla_{(\cdot)}\mathbf{L}$. Considering that the pair of coordinates (i, k) and (j, l) are independent and that for the EPI $i(t) = j(t) = t$, *i.e.* $\Delta i = \Delta j = \Delta t$, one obtains the relationship (7.10) defining $E_i(i, k) = L_i(i, j_n, k, l_m)$ and $E_k(i, k) = L_k(i, j_n, k, l_m)$.

An alternative gradient-based disparity estimation is obtained considering the concept of affine LF. A LF is denoted locally affine at $\mathbf{\Phi} = [i, j, k, l]^T$ if it is affine relatively to the coordinates of the ray in the image space within a neighborhood of $\mathbf{\Phi}$, *i.e.*

$$L\left(\mathbf{\Phi}\right) = \mathbf{A}^T \mathbf{\Phi} + b \tag{7.15}$$

with $\mathbf{A} = \left[a_i, a_j, a_k, a_l\right]^T$. One obtains a globally affine LF when the locally affine definition extends for the full domain of $\mathbf{\Phi}$. Note, however, that this is a too specific environment and camera setup. More in detail, in order to obtain a globally affine LF assumed to be smooth, it is required a planar scenario textured with a constant gradient which is imaged by a plenoptic camera orthogonal to the scene plane (Figure 7.7).

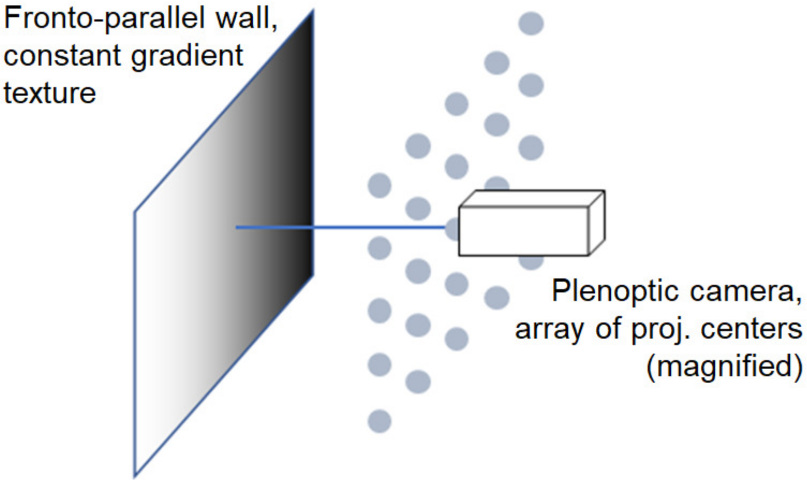


Figure 7.7: Setup to acquire a globally affine LF with a plenoptic camera. The array of circles represents the array of projection centers (not in scale) representing the viewpoint cameras of the plenoptic camera.

One way to obtain a locally affine LF is through a Taylor series expansion. Namely, making a first order Taylor expansion of the LF around the point $\mathbf{\Phi}_0$, one obtains

$$L\left(\mathbf{\Phi}\right) = L\left(\mathbf{\Phi}_0\right) + \nabla L\left(\mathbf{\Phi}_0\right)^T \left(\mathbf{\Phi} - \mathbf{\Phi}_0\right) + \text{H.O.T.} \tag{7.16}$$

where $\mathbf{\Phi}$ and $\mathbf{\Phi}_0 = \left[i_0, j_0, k_0, l_0\right]^T$ denote rays in the image space. If the higher order terms (H.O.T.) are discarded, one forms the locally affine LF (7.15) considering that $\mathbf{A} = \nabla L\left(\mathbf{\Phi}_0\right)$ and $b = L\left(\mathbf{\Phi}_0\right) - \nabla L\left(\mathbf{\Phi}_0\right)^T \mathbf{\Phi}_0$. This is a very interesting way of obtaining locally affine LFs as it allows considering many real LFs as opposed to the strict (unreal) theoretical example given for the globally affine LF.

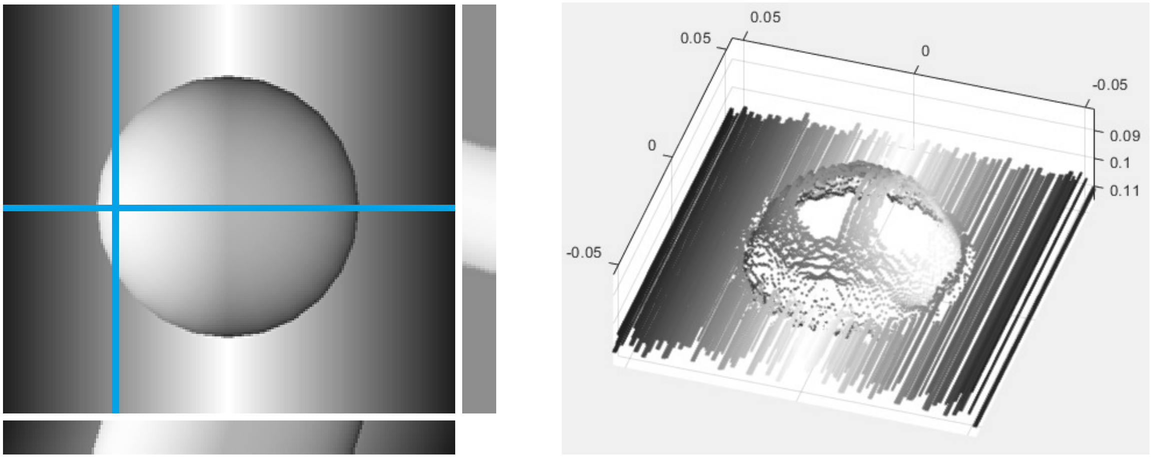
A LF characterized to be affine provides directly depth information [96]. Let us consider the globally affine LF example described previously consisting of a plane *colored* with a gradient (Figure 7.7). Considering the fronto-parallel plane Λ at depth $z = z_\Lambda$ with normal $\boldsymbol{\eta} = [0, 0, 1]^T$ such that a point $\mathbf{m} = [x, y, z_\Lambda]^T \in \Lambda \implies \mathbf{m}^T\boldsymbol{\eta} = z_\Lambda$. The color of the plane Λ at a point $\mathbf{m}$ is given by $c(\mathbf{m}) = \mathbf{m}^T\mathbf{g} + c_0$ where $\mathbf{g} = \left[g_x, g_y, 0\right]^T$ is the color gradient vector aligned with the plane Λ and c_0 is the color at point $\mathbf{m}_0 = [0, 0, z_\Lambda]^T$.

In order to obtain the color sampled in the image sensor, let us consider the parametric 3D line (3.11) associated with the ray $\boldsymbol{\Phi}$ in the image space and considering the LFIM (4.17). The intersection of the line with the plane Λ occurs at $\lambda = z_\Lambda$. This allows to define the plane color relatively to the coordinates of the ray in the image space, *i.e.* $c\left(\boldsymbol{\Phi}\right)$. The affine LF is then represented by (7.15) with $a_i = g_x\left(h_{si} + z_\Lambda\, h_{ui}\right)$, $a_j = g_y\left(h_{tj} + z_\Lambda\, h_{vj}\right)$, $a_k = g_x\, z_\Lambda\, h_{uk}$, $a_l = g_y\, z_\Lambda\, h_{vl}$, and $b = g_x\, z_\Lambda\, h_u + g_y\, z_\Lambda\, h_v + c_0$. The unknowns in this definition corresponds to the depth z_Λ, the color gradient vector $\mathbf{g}$ and c_0. Thus, to estimate the depth z_Λ, one can cancel the color gradient vector by dividing a_k with a_i or a_l with a_j. This allows to produce directly a depth estimate from the affine LF using

$$z_\Lambda = -\frac{h_{si}}{h_{ui} - \frac{a_i}{a_k}h_{uk}} \quad \text{and} \quad z_\Lambda = -\frac{h_{tj}}{h_{vj} - \frac{a_j}{a_l}h_{vl}} \tag{7.17}$$

which is equivalent to (7.11).

Let us consider an example of a locally affine LF consisting of a spherical hubcap on top of a plane with a gradient (Figure 7.8.a). In this case, the LF is not globally affine on the hubcap. The reconstruction using (7.17) is applied with the results illustrated in Figure 7.8.b. The results obtained show that this methodology can be used even on non-globally affine LFs since they are still locally affine, *i.e.* LFs are well represented locally by a first order Taylor series approximation. The mean of the absolute relative errors obtained is 1.49%.



(a)

(b)

Figure 7.8: Locally affine LF reconstruction. **(a)** Central VI of the scene surrounded by two EPIs. The bottom and right EPIs originate from the horizontal and vertical blue lines, respectively. **(b)** Reconstruction of the synthetic LF. Depth values are measured with respect to the camera coordinates frame.

Structure Tensor Estimation. Gradient of smoothed images can lead to cancellation effects and do not give reliable orientation information [39, 142]. The structure tensor is a more reliable descriptor of the local structure giving clues regarding edge and corner detection, and enabling the computation of orientations [16].

The gradient tensor $\mathbf{J}$ corresponds to the covariance matrix of the image gradients and is given at each pixel $\mathbf{n}$ as

$$\mathbf{J}(\mathbf{n}) = \nabla\mathbf{E}(\mathbf{n})\ \nabla\mathbf{E}(\mathbf{n})^T = \begin{bmatrix} E_i(\mathbf{n})\,E_i(\mathbf{n}) & E_i(\mathbf{n})\,E_k(\mathbf{n}) \\ E_i(\mathbf{n})\,E_k(\mathbf{n}) & E_k(\mathbf{n})\,E_k(\mathbf{n}) \end{bmatrix} \tag{7.18}$$

where $\nabla\mathbf{E}(\mathbf{n}) = \left[E_i(\mathbf{n}), E_k(\mathbf{n})\right]^T$ is the gradient vector of the EPI. For an easier notation, the dependency on $\mathbf{n}$ will not be represented in the following expressions.

The structure tensor is obtained by averaging a region of neighboring pixels instead of considering a single pixel. The averaging to obtain the structure tensor is necessary since the eigendecomposition of the gradient tensor (7.18) has only one non-zero eigenvalue that only enables edge identification. The spatial averaging considering a neighboring region of pixels allows to obtain two non-zero eigenvalues in regions where one has edges with different orientations. This enables to identify corners in the EPI [82], and therefore, occlusions and disocclusions.

The spatial averaging occurs in the estimation of the image gradient and for each of the components of the structure tensor. Therefore, the structure tensor $\mathbf{S}_{\tau\sigma}$ corresponds to the gradient tensor computed at pixel (i, k) by applying a Gaussian distribution with STD σ ($\mathbf{G}_\sigma$) to the EPI and then applying a Gaussian distribution with STD τ ($\mathbf{G}_\tau$) to each of the components of the gradient tensor

$$\mathbf{S}_{\tau\sigma} = \mathbf{G}_\tau * \mathbf{J}_\sigma \tag{7.19}$$

where $\mathbf{J}_\sigma = \nabla \mathbf{E}_\sigma \, \nabla \mathbf{E}_\sigma^T$, $\mathbf{E}_\sigma = \mathbf{G}_\sigma * \mathbf{E}$, $G_{(\cdot)} = \frac{1}{2\,\pi\,(\cdot)^2}\, e^{-\frac{\mathbf{n}^T\mathbf{n}}{2\,(\cdot)^2}}$, and $*$ denotes the convolution operator.

The structure tensor (7.19) corresponds to the structure tensor for a scalar valued image, for example, a grayscale image. If instead one has a vector valued image, for example, a colored image, the structure tensor is defined as a weighted sum of the structure tensors computed at each channel [26, 80, 142]. Thus, considering an image composed of N_c channels, the structure tensor is now defined as

$$\mathbf{S}_{\tau\sigma} = \sum_{c=1}^{N_c} w_c \, \mathbf{S}_{c,\tau\sigma} \tag{7.20}$$

where $\mathbf{S}_{c,\tau\sigma}$ is the structure tensor of the image channel c and that is computed using (7.19), and w_c is the weight associated with channel c. The sum of the weights for the N_c channels must be equal to one. Weickert *et al.* [142] considered that some channels can be more reliable than others and therefore can be weighted differently. Nonetheless, if there is no *a priori* knowledge of the noise one should assume an equal contribution for each of the channels.

Furthermore, since the structure tensor is the product of two filter results (image gradients), the Nyquist frequency is doubled. To avoid oversampling one should use a derivative filter band-limited by half of the limit frequency of the original image or interpolate the original image to duplicate its original resolution [83].

The principal directions of the local patch are determined from the eigendecomposition of the structure tensor. Denoting the structure tensor components as $\mathbf{S}_{\tau\sigma} = \begin{bmatrix} s_{11} & s_{12} \\ s_{12} & s_{22} \end{bmatrix}$, the eigenvalues are determined as

$$\lambda_{\max/\min} = \frac{1}{2}\left(s_{11} + s_{22} \pm \sqrt{(s_{11} - s_{22})^2 + 4\, s_{12}^2} \right) \quad , \tag{7.21}$$

and can be used to classify the regions of the image (Figure 7.9) as (i) homogeneous regions or regions without structure or texture ($\lambda_{\max} \approx 0$ and $\lambda_{\min} \approx 0$), (ii) edge regions *i.e.* regions with a dominant direction ($\lambda_{\max} > 0$ and $\lambda_{\min} \approx 0$), and (iii) corner regions *i.e.* regions with an ambiguous direction ($\lambda_{\max} > 0$ and $\lambda_{\min} > 0$).

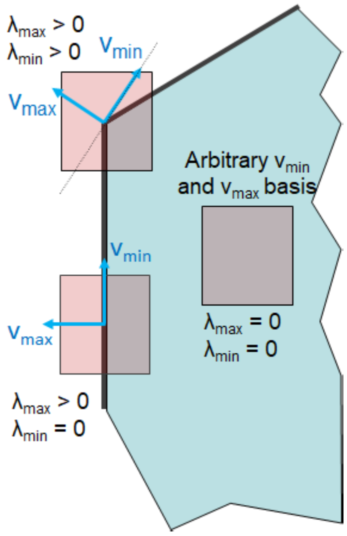


Figure 7.9: Regions according to structure tensor eigendecomposition.

The eigenvectors are determined by $v_{\max} = [\cos\alpha, \sin\alpha]^T$ and $v_{\min} = [-\sin\alpha, \cos\alpha]^T$ where α is the angle of the eigenvector associated with the maximum eigenvalue

$$\alpha = \frac{1}{2}\arctan\left(\frac{2\,s_{12}}{s_{11}-s_{22}}\right) \quad . \tag{7.22}$$

The slope of the lines in the EPI is obtained considering this angle by $\frac{\Delta k}{\Delta i} = \tan(\pi/2 - \alpha)$. This allows to obtain a depth estimate for each pixel of the EPI using (7.8).

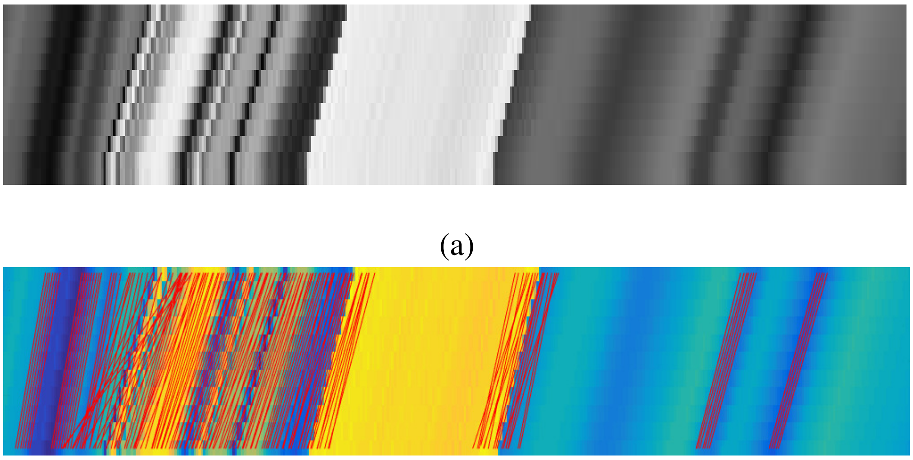
There are several measurements for the confidence of the structure tensor [10, 16, 26, 82, 142]. In the following, one considers the confidence measurement of Bigun et al. [16] that for each pixel $\mathbf{n}$ is given by

$$m = \begin{cases} \left(\frac{\lambda_{\max}-\lambda_{\min}}{\lambda_{\max}+\lambda_{\min}}\right)^2 & \text{if } \lambda_{\max}+\lambda_{\min} \neq 0 \\ 0 & \text{if } \lambda_{\max}+\lambda_{\min} = 0 \end{cases} \quad . \tag{7.23}$$

In Figures 7.11 and 7.10, one exhibits the disparity estimation using the methodology described. For example, in Figure 7.10, the structure tensor is used to estimate the slope of the lines in the EPI. The slopes are represented by lines in the figure and only estimates with confidence above a certain threshold are displayed. These figures show that this method allows to obtain noisy and sparse disparity estimates.

7.4 Cost-based Reconstruction

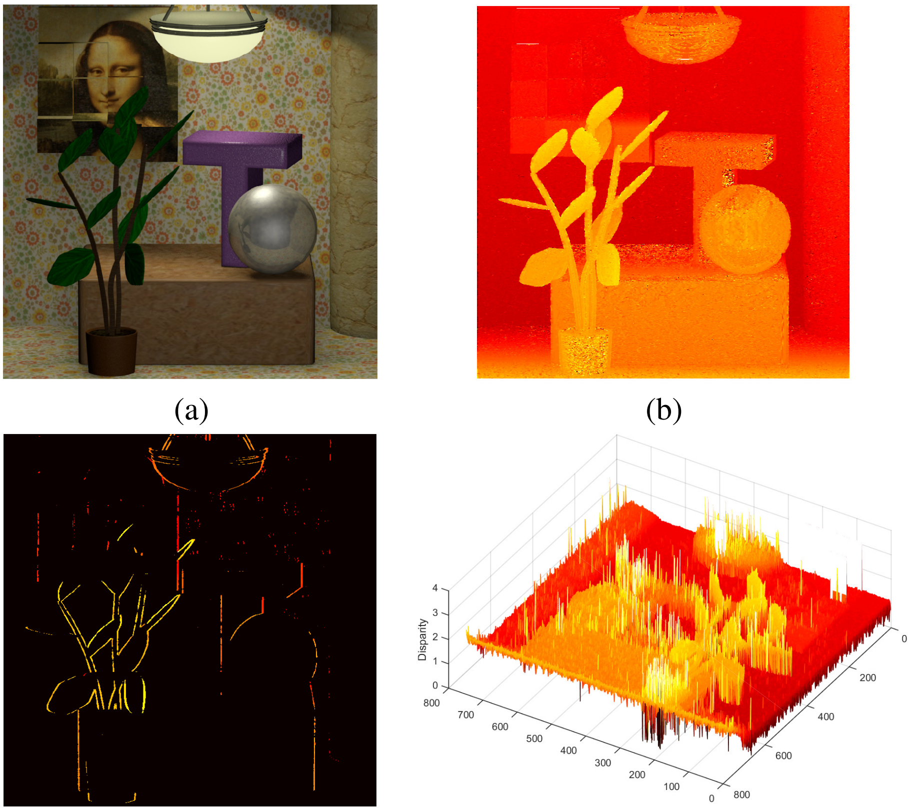
The gradient based approaches normally consider disparity estimates obtained from 2D slices of the LF (EPIs). Contrarily, the cost-based approaches analyze the rays in a SCam obtained from the 4D LF, as explained in Section 6.1. The disparity is then estimated based on minimizing or maximizing a metric that is applied to the content of the SCam for different pre-determined values of disparity.



(a)

(b)

Figure 7.10: Structure tensor disparity estimation. In **(a)**, an EPI taken from a synthetic LF is depicted. In **(b)**, the same EPI, with a different color scheme, and red lines marking the detected gradients is depicted. The color scheme still represents image intensities but it was chosen to make the gradients more visible.



(a)

(b)

(c)

(d)

Figure 7.11: Disparity estimates using the structure tensor analysis on each EPI of the LF. In **(a)**, one depicts the central viewpoint of the LF. The disparity map obtained from the structure tensor analysis is depicted in **(b)** and **(d)**. In **(c)**, only the disparity measurements with high confidence are displayed.

7.4.1 Photo-Similarity Metric for Affine LF

There are several metrics [76, 134] that can be used to evaluate the rays in a SCam but in this section one is going to focus in the correspondence metric proposed by Tao *et al.* [134]. This metric considers SCams defined by the microlens cameras after shearing the LF, *i.e.* the rays collected by these cameras follow the sampling (6.2). Hence, let us define the SCam $\mathbf{c}_q$ (6.2) associated with a reference ray $\mathbf{\Phi}_q = [i_r, j_r, k_r, l_r]^T$ considering an arbitrary disparity $\beta = \beta_{ik} = \beta_{jl}$ on the VIs as

$$c_q\left(i, j, \beta\right) = L\left(i_r + \Delta i, j_r + \Delta j, k_r + \beta \Delta i, l_r + \beta \Delta j\right) \tag{7.24}$$

where $\Delta i = i - i_r$ and $\Delta j = j - j_r$.

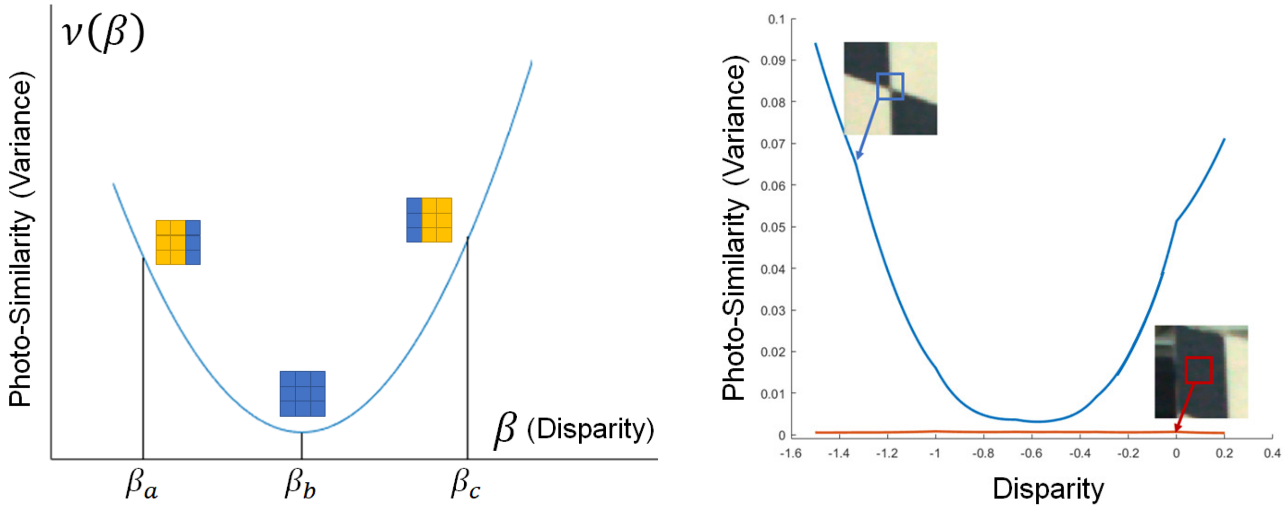
The correspondence metric [134] is evaluated comparing the intensity values collected on a SCam, namely, by evaluating the variance of the rays' intensities

$$\nu_q\left(\beta\right) = \frac{1}{N_i - 1}\frac{1}{N_j - 1}\sum_{i,j}\left[c_q\left(i, j, \beta\right) - \mu_q\left(\beta\right)\right]^2 \tag{7.25}$$

where

$$\mu_q\left(\beta\right) = \frac{1}{N_i N_j}\sum_{i,j} c_q\left(i, j, \beta\right) \tag{7.26}$$

corresponds to the mean intensity of the SCam, and N_i and N_j corresponds to the number of viewpoint in i- and j-direction. A correspondence candidate, and consequently a disparity candidate, is identified whenever the intensity of the rays in the SCam is similar, *i.e.* $\nu_q\left(\beta\right)$ is low (Figure 7.12.a). Nonetheless, for homogeneous regions the variance is also low (Figure 7.12.b), imagine for example the LF of a plenoptic camera imaging a plain white wall. In these regions, a low value is produced for $\nu_q\left(\beta\right)$ regardless of the disparity β being evaluated. Hence, the disparity that ends up being estimated is determined by the random noise present. In order to improve the quality of the disparity estimates, it is necessary to remove these values by thresholding a confidence measure associated to each estimate.



(a)

(b)

Figure 7.12: Photo-similarity metric. Illustration of typical results for the photo-similarity metric for different values of β **(a)** and in different regions of the VIs **(b)**. The red line corresponds to a pixel in a low gradient region while the blue line corresponds to a pixel in a high gradient region.

The minimum value of the variance $\nu_q(\beta)$ corresponds to the correct disparity of the ray $\mathbf{\Phi}_q$ whenever one has a locally affine LF (7.15). Let us define the SCam (7.24) using the definition of the affine LF obtained by a Taylor series expansion around the point $\mathbf{\Phi}_q$

$$\begin{aligned} c_q(i,j,\beta) \approx L\left(\mathbf{\Phi}_q\right) &+ \Big(L_i\left(\mathbf{\Phi}_q\right) + \beta L_k\left(\mathbf{\Phi}_q\right)\Big)\Delta i \\ &+ \Big(L_j\left(\mathbf{\Phi}_q\right) + \beta L_l\left(\mathbf{\Phi}_q\right)\Big)\Delta j \quad . \end{aligned} \tag{7.27}$$

This allows to represent the mean intensity of the SCam as

$$\mu_q(\beta) \approx L\left(\mathbf{\Phi}_q\right) + \gamma\Big(L_i\left(\mathbf{\Phi}_q\right) + \beta L_k\left(\mathbf{\Phi}_q\right)\Big) + \lambda\Big(L_j\left(\mathbf{\Phi}_q\right) + \beta L_l\left(\mathbf{\Phi}_q\right)\Big) \tag{7.28}$$

where $\gamma = \frac{1}{N_i}\sum_i \Delta i$ and $\lambda = \frac{1}{N_j}\sum_j \Delta j$. An imbalanced expansion around $\mathbf{\Phi}_q$ leads to $\gamma \neq 0$ and $\lambda \neq 0$ since the limits of the sum are defined as $i_{\min} = i_r - \delta$ and $i_{\max} = i_r + \delta + \kappa$ (equivalent for the coordinate j). Thus, considering $\omega = \frac{1}{N_i - 1}\frac{1}{N_j - 1}$, the photo-similarity metric is defined as

$$\begin{aligned}\nu_q(\beta) \approx \omega \sum_{i,j} \Big[&\Big(L_i\left(\mathbf{\Phi}_q\right) + \beta L_k\left(\mathbf{\Phi}_q\right)\Big)(\Delta i - \gamma) \\ &+ \Big(L_j\left(\mathbf{\Phi}_q\right) + \beta L_l\left(\mathbf{\Phi}_q\right)\Big)(\Delta j - \lambda)\Big]^2 \quad .\end{aligned} \tag{7.29}$$

From (7.10) and assuming the disparity is the same for (i, k) and (j, l), one has $\frac{L_i(\mathbf{\Phi}_q)}{L_k(\mathbf{\Phi}_q)} = \frac{L_j(\mathbf{\Phi}_q)}{L_l(\mathbf{\Phi}_q)}$. This allows to rewrite the variance as

$$\begin{aligned}\nu_q(\beta) \approx \omega \left(\beta + \frac{L_i\left(\mathbf{\Phi}_q\right)}{L_k\left(\mathbf{\Phi}_q\right)}\right)^2 \sum_{i,j} \Big[&L_k\left(\mathbf{\Phi}_q\right)(\Delta i - \gamma) \\ &+ L_l\left(\mathbf{\Phi}_q\right)(\Delta j - \lambda)\Big]^2 \quad .\end{aligned} \tag{7.30}$$

Analyzing the extrema, one can conclude that the minimum occurs at

$$\beta = -\frac{L_i\left(\mathbf{\Phi}_q\right)}{L_k\left(\mathbf{\Phi}_q\right)} \quad . \tag{7.31}$$

Hence, for an affine LF, $\nu_q(\beta)$ is a parabola with minimum at local disparity $\frac{\Delta k}{\Delta i}$ and curvature τ_q that increases with VI gradients.

$$\nu_q(\beta) = \tau_q \left(\beta - \frac{\Delta k}{\Delta i}\left(\mathbf{\Phi}_q\right)\right)^2 \tag{7.32}$$

where the curvature is defined as

$$\tau_q = \omega \sum_{i,j} \Big[L_k\left(\mathbf{\Phi}_q\right)(\Delta i - \gamma) + L_l\left(\mathbf{\Phi}_q\right)(\Delta j - \lambda)\Big]^2 \quad . \tag{7.33}$$

Analyzing $\nu_q\left(\beta\right)$ for pixels in areas with strong and weak gradients on the VIs (Figure 7.12.b), one can conclude that homogeneous regions produce flatter lines than textured regions as expected from the curvature. Hence, the curvature is a reasonable measure of confidence for the disparity estimates.

7.4.2 Disparity Refinement for Affine LF

The cost-based approach assumes pre-determined values of disparity to evaluate a metric. This leads to a discretization, *i.e.* a staircase effect (Figure 7.13.a), when there is a sub-sampling of the disparity range that lead to reconstructed points forming several constant depth planes corresponding to each disparity evaluated. This can be improved solving the location of the minimum of $\nu_q\left(\beta\right)$ with more precision by using the photo-similarity metric for an affine LF (7.32).

Considering the expansion of the photo-similarity metric (7.32), one has

$$\nu_q\left(\beta\right) = \tau_q\,\beta^2 + a_q\,\beta + b_q \tag{7.34}$$

where $a_q = 2\,\tau_q\,\frac{L_i\left(\mathbf{\Phi}_q\right)}{L_k\left(\mathbf{\Phi}_q\right)}$ and $b_q = \tau_q\left(\frac{L_i\left(\mathbf{\Phi}_q\right)}{L_k\left(\mathbf{\Phi}_q\right)}\right)^2$. Solving

$$\arg\min_{\beta}\ \nu_q\left(\beta\right) \quad , \tag{7.35}$$

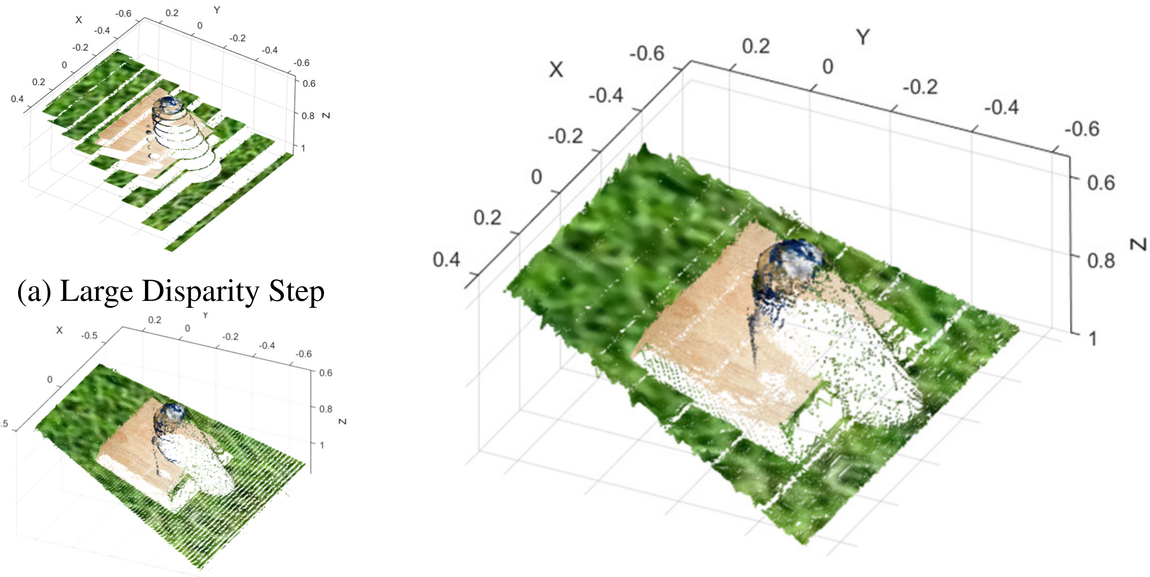
one obtains the following solution

$$\beta = -\frac{a_q}{2\,\tau_q} \quad . \tag{7.36}$$

Instead of using the information of the LF gradient to compute the disparity using (7.36), one can use the parameters $\mathbf{l} = \left[\tau_q, a_q, b_q\right]^T$ obtained by fitting the observations of $\nu_q\left(\beta\right)$ to a second order polynomial. Considering that the observation n is represented as $\mathbf{l}^T\mathbf{o}_n - \nu_q\left(\beta_n\right) = 0$ where $\mathbf{o}_n = \left[\beta_n^2, \beta_n, 1\right]^T$, the fitting parameters are obtained from the minimization problem

$$\arg\min_{\mathbf{l}} \sum_{n=1}^{N_\beta} \left(\mathbf{l}^T \mathbf{o}_n - \nu_q\left(\beta_n\right)\right)^2 \tag{7.37}$$

where N_β corresponds to the number of disparities evaluated. This problem can be solved using least-squares for example. The approach described allows to decrease the staircase effect while maintaining the same sampling for the disparity range (Figure 7.13.c).



(a) Large Disparity Step

(b) Small Disparity Step

(c) Large Disparity Step with Refinement

Figure 7.13: Depth estimation with different steps for the same disparity range. The estimate using a large disparity step is depicted in **(a)** while the estimate using a small disparity step is depicted in **(b)**. The staircase effect caused by a large disparity step is notorious in **(a)**. In **(c)**, one exhibits the improvement that can be made by applying the refinement proposed to **(a)**.

7.4.3 Refocusing for Affine LF

Disparity can also be estimated using defocus cues [134]. However, the disparity dependent information that can be retrieved from refocusing is limited. Namely, let us define the refocusing operation on a SCam as

$$r_q\left(\beta\right) = \frac{1}{N_i\, N_j} \sum_{i,j} c_q\left(i, j, \beta\right) \quad . \tag{7.38}$$

The refocusing operation corresponds to the mean intensity of the rays collected in the SCam $\mathbf{c}_q$, *i.e.* $r_q\left(\beta\right) = \mu_q\left(\beta\right)$. Consequently, for an affine LF, the refocusing is defined by (7.28).

A focal stack of an affine LF only conveys information that allows to estimate disparity if there is an imbalanced expansion around $\mathbf{\Phi}_q$, *i.e.* $\gamma \neq 0$ and $\lambda \neq 0$. Considering a balanced expansion around $\mathbf{\Phi}_q$, the refocusing operation reduces to $r_q\left(\beta\right) = L\left(\mathbf{\Phi}_q\right)$. In this case, the refocusing operation results in the same information regardless of the disparity used to perform the refocusing and no disparity-dependent information is present.

7.5 Dense Reconstruction

The disparity estimates obtained using the approaches described in Sections 7.2, 7.3 and 7.4 are noisy and sparse. For example, inspecting the disparity estimates using the structure tensor thresholded by the confidence measurements (Figure 7.11.c), one can see that the non-zero values correspond to edges and corners only. Therefore, in this section one presents a regularization framework for denoising and data fusion of the disparity estimates obtained from the LF.

7.5.1 Hypercube Representation

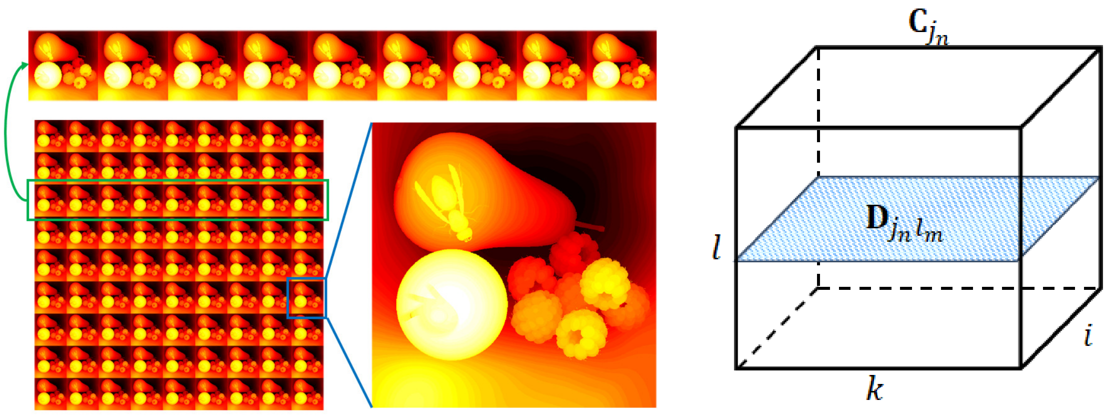
Let us consider the LF $\mathbf{L} \in \mathbb{R}^{N_i \times N_j \times N_k \times N_l}$ where N_i, N_j correspond to the number of viewpoint cameras in the i- and j-direction, and N_k, N_l to the number of pixels in the k- and l-direction. For each ray in the LF one can assign a disparity value using one of the previous approaches. Therefore, disparity has the same dimensionality as the LF, *i.e.* $\mathbf{D} \in \mathbb{R}^{N_i \times N_j \times N_k \times N_l}$.

Let us consider a representation for the disparity based on an hypercube $\mathbf{H}_{(.)}$. The hypercube is a set of datacubes $\mathbf{C}_{(.)}$ that are obtained fixing one of the viewpoint coordinates:

$$\mathbf{H}_j = \left\{ n \in \left\{1, \ldots, N_j\right\} : \mathbf{C}_{j_n}\left(i, k, l\right) = \mathbf{D}\left(i, j_n, k, l\right) \right\} \quad . \tag{7.39}$$

These datacubes $\mathbf{C}_{(.)}$ consist on a vertical stack of the disparity estimates obtained for each EPI (Figure 7.14). In the datacube representation, the disparity estimates for an EPI are obtained by fixing a pair of coordinates (i_n, k_m) or (j_n, l_m)

$$\mathbf{D}_{j_n, l_m}\left(i, k\right) = \mathbf{C}_{j_n}\left(i, k, l_m\right) = \mathbf{D}\left(i, j_n, k, l_m\right) \quad . \tag{7.40}$$



(a) (b)

Figure 7.14: Disparity hypercube representation. **(a)** Hypercube and datacube (green) represented as matrices, and VI (blue). **(b)** Datacube structure for fixed $j = j_n$.

Notice that the hypercube representation of the disparity observations can be obtained by fixing the other viewpoint coordinate i. Although this hypercube represents the same object, the disparity estimates and the confidence metric values may differ due to the nature of the disparity estimation. However, if one considers an hypercube representation of the LF, the information for the two hypercubes (obtained from fixing the different viewpoint coordinates) would be exactly the same.

7.5.2 Disparity Estimation Denoising

Let us consider that the disparity observations from a datacube $\mathbf{C}_{j_n}$ can be represented as a two-dimensional matrix where each line corresponds to the disparity retrieved from each of the pixels of the EPI, lexicographically ordered (Figure 7.14). Let the matrix representing the observed disparity be $\mathbf{Y}_{j_n} \in \mathbb{R}^{N_l \times (N_i \times N_k)}$. Assuming that the observations are only affected by i.i.d. additive noise $\mathbf{W}_{j_n} \in \mathbb{R}^{N_l \times (N_i \times N_k)}$, one can model the disparity observations as

$$\mathbf{Y}_{j_n} = \mathbf{Z}_{j_n} + \mathbf{W}_{j_n} \tag{7.41}$$

where $\mathbf{Z}_{j_n} \in \mathbb{R}^{N_l \times (N_i \times N_k)}$ are the real disparities from the datacube. For simplicity, one assumes that the boundary conditions are periodic. This allows to compute convolutions and matrix inversions using Fast Fourier Transforms (FFTs), see more details in Appendix F. The observation model (7.41) can be generalized for the hypercube $\mathbf{H}_j$ by including the datacubes $\mathbf{C}_{j_n}$ for $n = 1, \ldots, N_j$:

$$\mathbf{Y}_j = \mathbf{Z}_j + \mathbf{W}_j \tag{7.42}$$

with $\mathbf{Y}_j$, $\mathbf{Z}_j$, $\mathbf{W}_j \in \mathbb{R}^{(N_j \times N_l) \times (N_i \times N_k)}$ resulting from the vertical stack of the matrices $\mathbf{Y}_{j_n}$, $\mathbf{Z}_{j_n}$, and $\mathbf{W}_{j_n}$ for $n = 1, \ldots, N_j$, respectively

$$\mathbf{Y}_j = \begin{bmatrix} \mathbf{Y}_{j_1} \\ \cdots \\ \mathbf{Y}_{j_{n_j}} \end{bmatrix}, \quad \mathbf{Z}_j = \begin{bmatrix} \mathbf{Z}_{j_1} \\ \cdots \\ \mathbf{Z}_{j_{n_j}} \end{bmatrix}, \quad \mathbf{W}_j = \begin{bmatrix} \mathbf{W}_{j_1} \\ \cdots \\ \mathbf{W}_{j_{n_j}} \end{bmatrix} \quad . \tag{7.43}$$

This allows to include the disparity observations from several viewpoints while considering the same virtual camera motion to obtain the EPIs.

In Figure 7.14 there is an example of the matrices $\mathbf{Z}_j$and $\mathbf{Z}_{j_n}$. These structures represent a repeating sequence of disparity images that differ in a small number of pixels due to the different viewpoint coordinates.

Since one obtains natural like images one applies an isotropic total variation regularizer proposed by Rudin *et al.* [126] to promote sharp discontinuities at edges. This type of regularizer is regularly used in image denoising [28] and has already been used in the context of LF analysis [56, 139]. This leads to the following optimization problem:

$$\arg\min_{\mathbf{Z}_j} \left(\frac{1}{2} \|\mathbf{Y}_j - \mathbf{Z}_j\|_F^2 + \lambda_r \mathrm{TV}\left(\mathbf{Z}_j\mathbf{D}_h, \mathbf{Z}_j\mathbf{D}_v\right) \right) \tag{7.44}$$

where $\|\cdot\|_F = \sqrt{\mathrm{tr}\left((\cdot)(\cdot)^T\right)}$ corresponds to the Frobenius norm, TV corresponds to the total variation regularizer that is given by

$$\mathrm{TV}\left(\mathbf{Z}_j\mathbf{D}_h, \mathbf{Z}_j\mathbf{D}_v\right) = \sum_{n,m} \mathrm{TV}_{n,m}\left(\mathbf{Z}_j\mathbf{D}_h, \mathbf{Z}_j\mathbf{D}_v\right) \tag{7.45}$$

$$\mathrm{TV}_{n,m}\left(\mathbf{Z}_j\mathbf{D}_h, \mathbf{Z}_j\mathbf{D}_v\right) = \sqrt{\left[\left(\mathbf{Z}_j\mathbf{D}_h\right)_{n,m}\right]^2 + \left[\left(\mathbf{Z}_j\mathbf{D}_v\right)_{n,m}\right]^2} \tag{7.46}$$

where $(\cdot)_{n,m}$ corresponds to the entry (n, m) of matrix $(\cdot)$, and $\mathbf{D}_h$ and $\mathbf{D}_v$ are operators for the horizontal and vertical finite differences considering periodic boundary conditions, respectively. The solution to this optimization problem is given by the method of ADMM and the details are presented in Section 7.5.3 and Appendix F.

7.5.3 Disparity Estimation Data Fusion

The LF allows to obtain two different hypercube representations that simulate two types of virtual linear paths that can be used to retrieve the EPIs. The denoising problem presented in Section 7.5.2 considered only one of the possible observations of the hypercube. The approach described in this section allows to integrate the information of the disparities obtained using the two virtual linear paths for obtaining the EPIs.

Let us consider the disparity observations from the hypercubes corresponding to fixing the viewpoint coordinate j as $\mathbf{Y}_j$ and to fixing the viewpoint coordinate i as $\mathbf{Y}_i$. The two observations of the hypercube are related with each other by $\mathbf{Y}_i = \mathbf{Y}_j^T$, assuming that the observations are not affected by noise. From Section 7.5.2, one knows that $\mathbf{Y}_j \in \mathbb{R}^{\left(N_j \times N_l\right) \times (N_i \times N_k)}$ and, consequently, $\mathbf{Y}_i \in \mathbb{R}^{(N_i \times N_k) \times \left(N_j \times N_l\right)}$. Hence, let us consider a matrix $\mathbf{Z} \in \mathbb{R}^{\left(N_j \times N_l\right) \times (N_i \times N_k)}$ that corresponds to the real disparity map to be estimated and that integrates the information of the two observations.

The observation models for the hypercube for each of the camera paths is given by:

$$\mathbf{Y}_j = \mathbf{M}_j \, \mathbf{Z} + \mathbf{W}_j \quad \text{and} \quad \mathbf{Y}_i = \mathbf{M}_i^T \mathbf{Z}^T + \mathbf{W}_i \tag{7.47}$$

where $\mathbf{M}_j \in \mathbb{R}^{\left(N_j \times N_l\right) \times \left(N_j \times N_l\right)}$, and $\mathbf{M}_i \in \mathbb{R}^{(N_i \times N_k) \times (N_i \times N_k)}$ represent a uniform sub-sampling of $\mathbf{Z}$ to obtain the disparity information that correspond to each of the camera motions. $\mathbf{M}_j$ has lines and $\mathbf{M}_i$ has columns that are a subset of the columns of the identity matrix. $\mathbf{W}_j \in \mathbb{R}^{\left(N_j \times N_l\right) \times (N_i \times N_k)}$, and $\mathbf{W}_i \in \mathbb{R}^{(N_i \times N_k) \times \left(N_j \times N_l\right)}$ represent i.i.d. additive noise. Similarly, to the denoising problem one considers the total variation (7.46) as the regularizer. The optimization problem can then be formalized using the observation models and the regularizer as

$$\arg\min_{\mathbf{Z}} \left(\frac{1}{2} \left\| \mathbf{Y}_j - \mathbf{M}_j \, \mathbf{Z} \right\|_F^2 + \frac{\lambda_i}{2} \left\| \mathbf{Y}_i^T - \mathbf{Z} \, \mathbf{M}_i \right\|_F^2 + \lambda_r \mathrm{TV} \left(\mathbf{Z}\mathbf{D}_h, \mathbf{Z}\mathbf{D}_v \right) \right) . \tag{7.48}$$

In this optimization problem, the first two data terms are data-fitting terms while the last is the regularizer. The data terms explain the observed disparities considering the observation models for $\mathbf{Y}_j$ and $\mathbf{Y}_i$ (7.47). The weights λ_i and λ_r allow to control the contribution of each of the terms. This optimization problem is similar to the data fusion problem presented by Simões et al. [131] for hyperspectral cameras superresolution. In this problem, one faces the same problems indicated by Simões et al. [131]: (i) high dimensionality of the variable to be estimated $\mathbf{Z}$, (ii) the regularizer is non-quadratic and non-smooth although the optimization problem (7.48) is convex, and (iii) the sampling operators do not allow to use the FFT directly.

Therefore, one follows the same approach presented in [131]. Namely, by using an ADMM instance, the Split Augmented Lagrangian Shrinkage Algorithm (SALSA) [4]. Thus, the optimization variable $\mathbf{Z}$ is split into auxiliary variables using the variable splitting technique to be able of applying the ADMM method. The optimization problem (7.48) is now defined by

$$\begin{gathered}\arg \min_{\mathbf{Z},\mathbf{V}_1,\mathbf{V}_2,\mathbf{V}_3,\mathbf{V}_4} \left(\frac{1}{2} \left\| \mathbf{Y}_j - \mathbf{M}_j \, \mathbf{V}_1 \right\|_F^2 + \frac{\lambda_i}{2} \left\| \mathbf{Y}_i^T - \mathbf{V}_2 \, \mathbf{M}_i \right\|_F^2 + \lambda_r \mathrm{TV} \left(\mathbf{V}_3, \mathbf{V}_4 \right) \right) \\ \text{s.t.} \quad \mathbf{V}_1 = \mathbf{Z}, \quad \mathbf{V}_2 = \mathbf{Z}, \quad \mathbf{V}_3 = \mathbf{Z}\mathbf{D}_h, \quad \mathbf{V}_4 = \mathbf{Z}\mathbf{D}_v \quad . \end{gathered} \tag{7.49}$$

Considering

$$\mathrm{f}(\mathbf{V}) = \frac{1}{2} \left\| \mathbf{Y}_j - \mathbf{M}_j \, \mathbf{V}_1 \right\|_F^2 + \frac{\lambda_i}{2} \left\| \mathbf{Y}_i^T - \mathbf{V}_2 \, \mathbf{M}_i \right\|_F^2 + \lambda_r \mathrm{TV} \left(\mathbf{V}_3, \mathbf{V}_4 \right) \tag{7.50}$$

and

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_1^T \\ \mathbf{V}_2^T \\ \mathbf{V}_3^T \\ \mathbf{V}_4^T \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \mathbf{I} \\ \mathbf{I} \\ \mathbf{D}_h^T \\ \mathbf{D}_v^T \end{bmatrix} \quad , \tag{7.51}$$

one can rewrite the optimization problem (7.49) as

$$\arg \min_{\mathbf{Z},\mathbf{V}} \quad \mathrm{f}\left(\mathbf{V}\right) \quad \text{s.t.} \quad \mathbf{V} = \mathbf{G}\,\mathbf{Z}^T \quad . \tag{7.52}$$

This problem has the following Augmented Lagrangian [114], disregarding the part that only depends on the scaled dual variable $\mathbf{A}$

$$\mathcal{L}(\mathbf{Z}, \mathbf{V}, \mathbf{A}) = \mathrm{f}\left(\mathbf{V}\right) + \frac{\mu}{2} \left\| \mathbf{G}\, \mathbf{Z}^T - \mathbf{V} - \mathbf{A} \right\|_F^2 \tag{7.53}$$

where μ is a positive constant called the penalty parameter. Now, one is able to apply the ADMM method, more specifically SALSA [4] as summarized in Algorithm 3.

```
Algorithm 3: Disparity Data Fusion
Input  : Observations: $\mathbf{Y}_j$, $\mathbf{Y}_i$
         Parameters: $\lambda_i$, $\lambda_r$, $\mu$
         Initializations: $\mathbf{V}^{(0)}$, $\mathbf{A}^{(0)}$
Output: LF Disparity Map: $\mathbf{Z}$
repeat
    $\mathbf{Z}^{(k+1)} \in \arg\min_{\mathbf{Z}} \; \mathcal{L}(\mathbf{Z}, \mathbf{V}^{(k)}, \mathbf{A}^{(k)})$
    $\mathbf{V}^{(k+1)} \in \arg\min_{\mathbf{V}} \; \mathcal{L}(\mathbf{Z}^{(k+1)}, \mathbf{V}, \mathbf{A}^{(k)})$
    $\mathbf{A}^{(k+1)} = \mathbf{A}^{(k)} - \left(\mathbf{G}\, \mathbf{Z}^{T(k+1)} - \mathbf{V}^{(k+1)}\right)$
until stopping criterion is satisfied;
```

The conditions for the convergence of SALSA are established in [4]. The algorithm requires that $\mathbf{G}$ must have full column rank and the function $\mathrm{f}\left(\cdot\right)$ must be closed, proper, and convex. The algorithm described above has a matrix $\mathbf{G}$ with full column rank (due to the presence of identity matrix $\mathbf{I}$), and the function $\mathrm{f}\left(\cdot\right)$ is a sum of closed, proper, and convex functions. Therefore, the conditions for the convergence of the algorithm are met. The detailed minimization problems can be found in Appendix F.

The solution for the denoising problem in Section 7.5.2 is achieved by considering only one of the data terms, $\mathbf{Y}_j$, and that the sub-sampling matrix is equal to the identity matrix ($\mathbf{M}_j = \mathbf{I}$) in the optimization problem (7.48). Therefore, the minimization problem regarding the auxiliary variable $\mathbf{V}_2$ disappears, and the minimization problem regarding the disparity map $\mathbf{Z}$ is simplified. This optimization problem can also be solved using a primal-dual solver, for example using the Forward Backward Primal Dual algorithm of Condat *et al.* [34].

7.5.4 Experimental Results

The methodology described for dense reconstruction is applied to the *Still* synthetic dataset provided by the HCI Heidelberg Group [140] (Figure 7.15). The results of the regularization with one and two data terms are compared with the disparity estimates from the structure tensor. The ground truth provided with the synthetic dataset is used to determine the PSNR values.

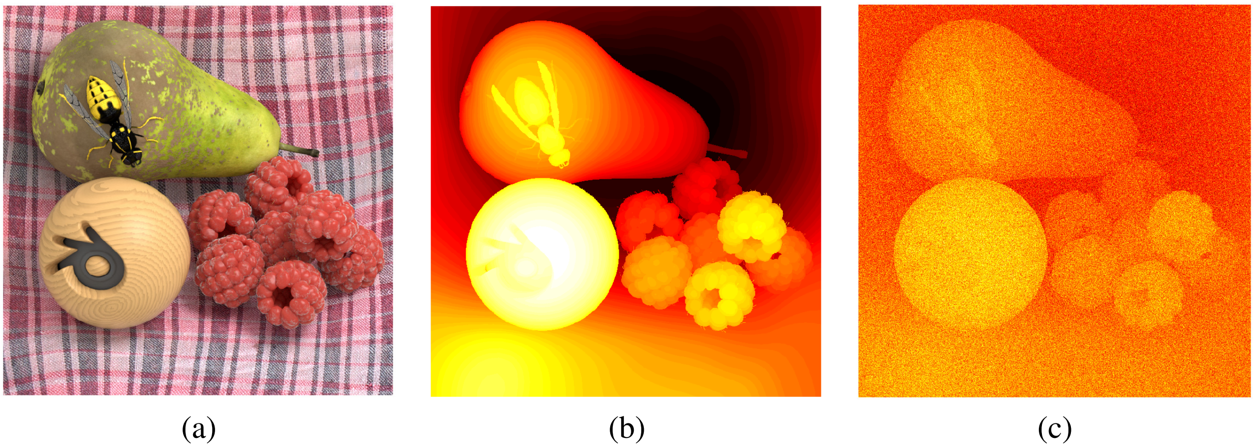
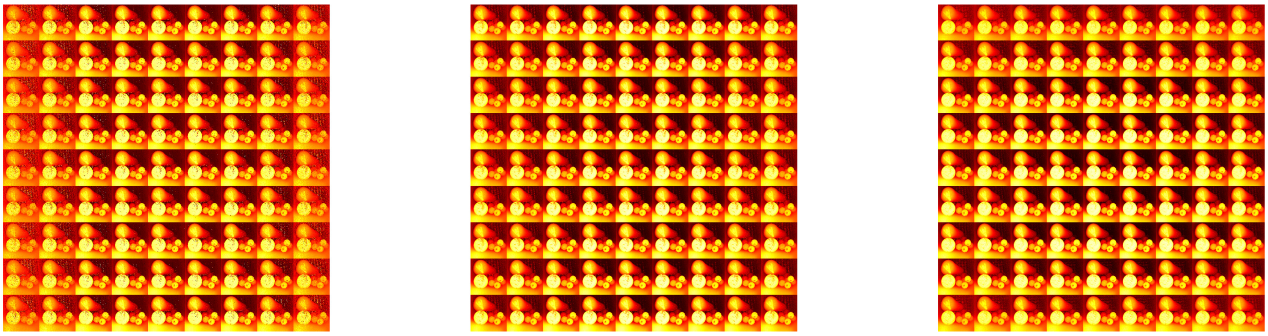


Figure 7.15: *Still* live dataset of the HCI Heidelberg Group [140]. The central VI is displayed on **(a)**. The disparity ground truth information is depicted on **(b)** while the generated observed disparity is displayed on **(c)**.

The structure tensor analysis is implemented as described in Section 7.3 assuming an equal contribution for each of the color channels since one does not have *a priori* knowledge of the noise [142]. The structure tensor is computed using an upsampled version of the EPIs to compensate for the doubled Nyquist frequency [83]. Furthermore, the smoothing of the image and the components of the structure tensor is obtained by applying Gaussian distributions with STD 0.8 and 3.2, respectively, following the suggestion of Köethe [82]. For the optimization problem, one considers an equal contribution for each of the data terms ($\lambda_i = 1$), and the penalty parameter μ to be fixed and equal to 1 since it only affects the convergence speed and not the convergence. The parameter λ_r is fine-tuned by performing a denoising of the disparity ground truth for several values of λ_r ranging from 2^{-20} to 2^7 and selecting the one that provides the highest PSNR. The total number of iterations is fixed and equal to 20.

From Figure 7.16, one can see that the hypercube obtained from the structure tensor analysis presents a noticeable decay on accuracy in the peripheral viewpoints. Focusing on a specific viewpoint (peripheral or central) in Figure 7.17, one can conclude that the disparity estimates are less accurate on homogeneous regions which in the EPI represent regions of constant intensity between the lines that one wants to detect. A small change of intensity in these regions lead to disparity estimates that change rapidly and have high variability. This is more noticeable on the 3D representation of the disparity estimates (Figure 7.17.b and 7.17.d).



(a) Structure Tensor

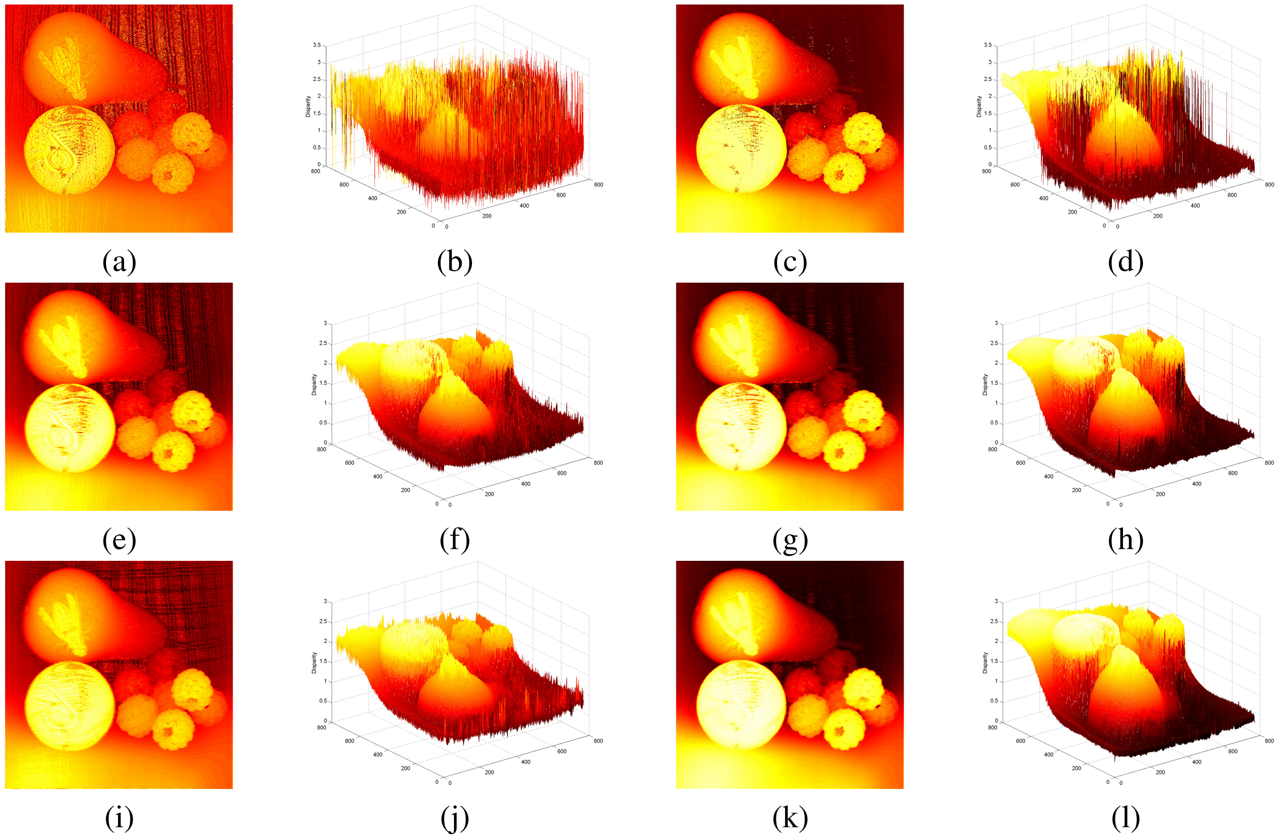
(b) 1D Regularized Structure Tensor

(c) 2D Regularized Structure Tensor

Figure 7.16: Hypercube disparity estimation before **(a)** and after regularization considering one **(b)** and two **(c)** observations of the hypercube.

The hypercube after the regularization has reduced noise and the accuracy remains almost the same from the central to the peripheral viewpoints. This is confirmed by the increased PSNR after regularization (from 8.65 dB to 10.76 dB). The noise can be further reduced by considering the additional data term of the optimization problem (7.44) (PSNR of 11.00 dB). The usage of the two disparity data terms allows to improve the estimates since the estimated hypercube corresponds to a compromise between the two disparity observations of the hypercube.

Additionally, the formulation allows to select specific disparity estimates for each of the observations of the hypercube through the sub-



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

(e) (f) (g) (h)

(i) (j) (k) (l)

Figure 7.17: Disparity estimation using the structure tensor analysis (top) and after regularization with one (middle) and two data terms (bottom). The columns correspond to the disparity estimates using a 2D representation (1^{st} and 3^{rd} columns) and a 3D representation (2^{nd} and 4^{th} columns) to highlight the noise. Peripheral viewpoints are depicted on the 1^{st} and 2^{nd} columns. Central viewpoints are depicted on the 3^{rd} and 4^{th} columns.

sampling matrices $\mathbf{M}_{(\cdot)}$. Therefore, one performed the same analysis but now considering only the disparity observations with higher confidence values from each hypercube observation. In this scenario, a compromise between the two hypercube observations will only occur for disparities with similar confidence values between the two observations instead of achieving this compromise for all disparities. Hence, this approach leads to an increase in the PSNR value for 11.38 dB.

7.6 Deep Neural Network Reconstruction

Deep neural networks generally require intense training, and while they may lead to good results, they may also result in an inability to perform well under inputs with characteristics outside their training scope. Normally, one can augment the training data [130] for the network to generalize for other inputs or one can perform transfer learning of the neural network [47]. However, this is not always possible due to constraints such as lack of data, time or computational power.

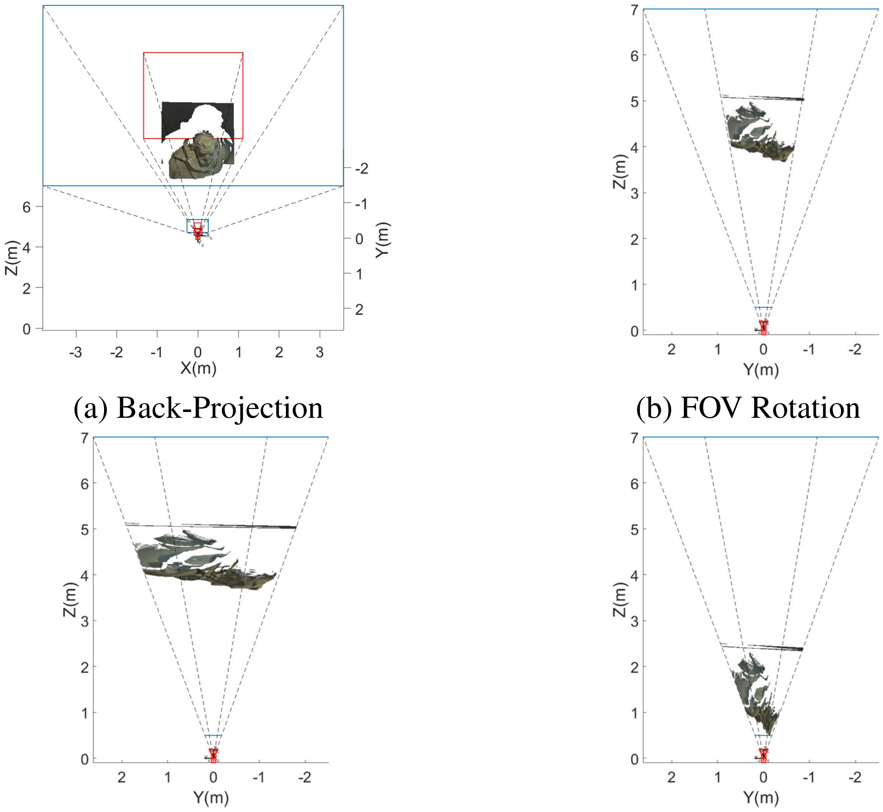
In the DDFFNet from Hazirbas *et al.* [65], the focal stack used has implicit a specific camera geometry defined by the microlens array and the main lens configuration (zoom and focus) of the plenoptic camera used to acquire the LF dataset. More specifically, the DDFFNet requires a focal stack of ten images each focused at linearly spaced disparities covering a disparity range of $[0.02, 0.28]$ pixels. The disparity range for the camera configuration used corresponds to a depth range of $[0.5, 7.0]$ meters. Let us denote the camera configuration used by Hazirbas *et al.* [65] as the source camera configuration.

Cameras with different characteristics from the source camera will capture the same scene in a different way and produce images with different information. As an example, the scene disparity range shifts if the main lens focal plane changes or the disparity range increases if one considers a higher baseline. This can lead to disparity values that were not considered during training. Also, changing the depth of the scene or the disparities considered to obtain the focal stack will also lead to an inability to correctly reconstruct the scene. Let us denote the camera with different characteristics from the source camera as the target camera. In this section, one extends the network application range of Hazirbas *et al.* [65] by accepting larger input disparity ranges that can be obtained considering different scenes or camera configurations.

7.6.1 Target-Source-Target LF Mapping

Let us consider the camera array representation for a plenoptic camera (Section 4.1). The FOV of a plenoptic camera is bounded by the envelope of all cameras' fields of view (pyramids) in the array. This envelope corresponds to the pyramid of the central viewpoint in the main lens focal plane and is not much wider than this pyramid for other planes due to the small baselines (Section 4.1.3). Therefore, in these transformation one is going to consider only the central viewpoint.

The method proposed extends the application range by transforming the input LF so that it falls under the training conditions. This mapping is achieved by back-projecting the target LF into a point cloud, transforming the generated point cloud and reprojecting the scenery onto an array of cameras identical to the source camera to obtain the corresponding LF (Figure 7.18) [122].



(a) Back-Projection

(b) FOV Rotation

(c) FOV Scaling

(d) Depth Scaling

Figure 7.18: Target-Source LF mapping steps for the Cotton dataset [68]. The disparity range of this dataset is $[-1.6, 1.5]$ pixels which is outside the disparity range of the source camera. **(a)** exhibits the point cloud obtained from back-projecting the central viewpoint rays with a given disparity map. The FOV of the target camera is displayed in red while the FOV of the source camera is displayed in blue. **(b)** highlights the alignment of the point cloud with the principal axis of the camera. **(c)** and **(d)** show the FOV and depth scaling that is done in order to ensure the scene geometry uses a wider FOV and depth range within the source camera available ranges, respectively.

Back-Projection. In the first step, the rays corresponding to the central viewpoint camera are back-projected, resulting in a 3D point cloud

(Figure 7.18.a). The back-projection requires a calibrated camera and a disparity map for the central VI. In the example provided in Figure 7.18, one used the ground truth provided with the dataset [68] but in a real scenario one should estimate disparity using one of the approaches described in the previous sections. The disparity map, having a calibrated camera, is easily converted to a depth map using (7.8). Then, the (x, y) coordinates of each point in the point cloud for a known depth z is obtained by the back-projection (3.11) with (i, j) fixed to the central viewpoint coordinates. For each point of the point cloud, one stores the corresponding intensity value.

FOV Rotation. The FOV for the target and source cameras are obtained by back-projecting the viewpoint camera limit pixels (Figure 7.18.a) The generated point cloud is rotated around the optical center along the x- and y- axis in order to align the target and source principal axis (Figure 7.18.b). The rotation angles are computed using a point at depth z in each of the principal axis, namely, $\theta = \tan^{-1}\left(\frac{(\cdot)_t - (\cdot)_s}{z}\right)$ with $(\cdot)_t$ and $(\cdot)_s$ denoting the point x- or y- coordinate on the target and source principal axis, respectively.

FOV Scaling. In order to use a wider FOV of the source camera, the (x, y) coordinates of the point cloud are scaled by the same factor to avoid distortion. Hence, the scaling factor is the one that scales the point cloud so that it matches the source smaller FOV side (Figure 7.18.c).

Depth Scaling. In this step, one scales the z-coordinate of the point cloud such that the point cloud spans a wider depth range and is within the source depth range. Inspecting the supplementary material in [65], one concludes that using a smaller range, depth range [0.5, 2.5] meters, results in a more well distributed set of focused depths. The scaling to the new depth range $[z_1', z_2']$ is obtained by an affine transformation whose parameters are obtained from the solution of the linear system

$$\begin{bmatrix} z_m & 1 \\ z_M & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} z'_1 \\ z'_2 \end{bmatrix} \tag{7.54}$$

with z_m and z_M corresponding to the minimum and maximum depth of the point cloud. The (x, y) coordinates are recomputed in order to ensure that the point cloud is within the source plenoptic camera pyramid. The new coordinates are obtained by multiplying the coordinates of the point by $\frac{z'}{z}$ where z' is the new z-coordinate after the depth scaling.

Reprojection. The final step corresponds to a reprojection of the point cloud into the camera array of the source camera used by Hazirbas *et al.* [65]. This allows to obtain a new LF which will then be refocused and used to create the input focal stack for the DDFFNet. This LF is within the range of the network training conditions, so one expects to obtain better depth estimates than using the LF of the target camera without any transformations.

The DDFFNet is used to obtain a new depth map considering the source camera configuration. Then, this depth map is used to obtain a new point cloud using the source camera calibration. In this case, for each point on the point cloud one stores the corresponding depth. The generated point cloud is then transformed using the inverse transformation of each step described previously, in reverse order. In the final step, the depth of each point of the point cloud is reprojected onto the central viewpoint camera of the target camera, forming a depth map. The depth map is then converted to a disparity map using (7.8).

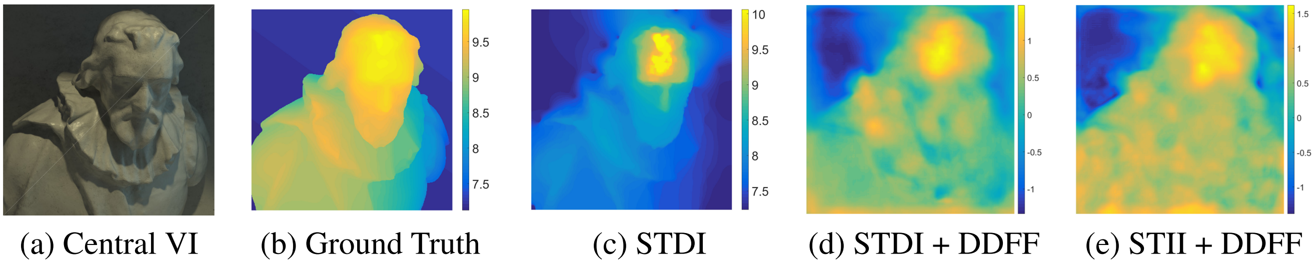
7.6.2 Experimental Results

The proposed method is evaluated on the Training Set of the 4D LF Benchmark [68], the same used by Hazirbas *et al.* [65] to evaluate the network performance after retraining. The disparity maps obtained can be compared with the ground truth provided by these datasets [68].

In a LF acquired by a plenoptic camera, the depth map is not available so one has to first estimate a depth map. In this experiment, an initial disparity map is obtained using the structure tensor (Section 7.3.1). The estimated disparity map is sparse, hence, one proposes two strategies for obtaining a dense map: (i) converting disparity to depth, constructing the point cloud as is and inpaint each source VI in intensities (STII), or (ii) inpainting the disparity map, converting disparity to depth and then generating a full point cloud (STDI). The results of applying DDFFNet without transforming the target datasets or using the STDI and STII approaches are summarized in Table 7.2. The ground truth based approach to guide the transformation of the target datasets is used to show the best outcome one can obtain. The numerical results are presented as defined in [65]. As a qualitative analysis, the disparity map obtained by each method is presented in Figure 7.19.

Method	Target LF	Retrain	Ground Truth	STDI	STII + DDFF	STDI + DDFF
Disparity MSE	0.7741	0.19	0.3002	0.7383	0.5378	0.3392
Disparity RMS	0.8709	0.42	0.5463	0.7934	0.7227	0.5765
Depth MSE	0.9395	—-	0.2934	1.1063	0.6499	0.3104
Depth RMS	0.7233	—-	0.4220	0.6950	0.5958	0.4332

Table 7.2: Disparity and depth MSE and RMS errors. The second column exhibits the results of applying DDFFNet directly to the target LFs. The results obtained by retraining the DDFFNet (retrieved from [65]) and applying the structure tensor on the target LFs followed by inpainting are displayed on the third and fifth columns. The results obtained using the ground truth to perform the transformation from the target to source camera is presented in the fourth column. The results of the STDI and STII approaches to guide the transformation from the target to source camera are presented on the last two columns.



(a) Central VI (b) Ground Truth (c) STDI (d) STDI + DDFF (e) STII + DDFF

Figure 7.19: Depth reconstruction for Cotton dataset [68]. LF central VI **(a)** and ground truth depth map **(b)**. Depth reconstruction using the structure tensor considering inpainting of pixel intensity values (STII) or disparity values (STDI) **(c)**-**(e)**.

Table 7.2 shows that applying the network directly on the target LFs produces a high depth error. However, applying the proposed method using the STDI approach, one is able to reduce the error in almost 40% which shows the validity of the approach.

7.7 Chapter Summary

In this chapter, was proposed an improved metric reconstruction methodology based on multiview stereo for a calibrated plenoptic camera by imposing geometry cues on the ray-spaces (i, k) and (j, l) (Section 7.2).

The gradient-based approaches to recover depth from the LF are reviewed to consider the EPI geometry of plenoptic cameras defined in Sections 4.1.3 and 5.1.3. In Section 7.3, one also derives the disparity estimation from LF gradients by using the concept of optical flow and the original concept of affine LF.

The affine LF concept is proposed in the context of cost-based approaches to gain some insights on the metrics normally used (Section 7.4). Namely, it is derived the expression of the correspondence metric [134] for an affine LF. The correspondence metric is a parabola with a minimum at the disparity value and with curvature that depends on the VI gradients. This expression allows to define a methodology for disparity improvement and identify that the curvature can be used as a confidence metric. Additionally, one shows that a focal stack of an affine LF hardly exhibits disparity-dependent information.

A data fusion problem which uses the full 4D LF by considering disparity estimates for each 2D EPI of the hypercube is formalized to recover a dense disparity map (Section 7.5). The optimization problem was solved by resorting to an ADMM instance that provides good results with few iterations. Furthermore, were considered periodic boundary conditions that allowed to use FFTs in the computations allowing the algorithm to be computationally efficient.

The network application range of Hazirbas *et al.* [65] is extended in order to accept larger input disparity ranges that can be obtained considering different scenes or camera configurations. The method presented extends the application range by transforming the input LF so that it falls under the training conditions. The proposed method provides a faster and more versatile approach at the cost of loosing some accuracy relatively to a full retraining approach.

Chapter 8

Conclusions and Future Work

In this chapter, contrarily to the organization of the thesis that presented a plenoptic camera type-based view of the work, are presented the major conclusions regarding the contributions identified in Chapter 1.

8.1 Unification of Geometric Projection Models

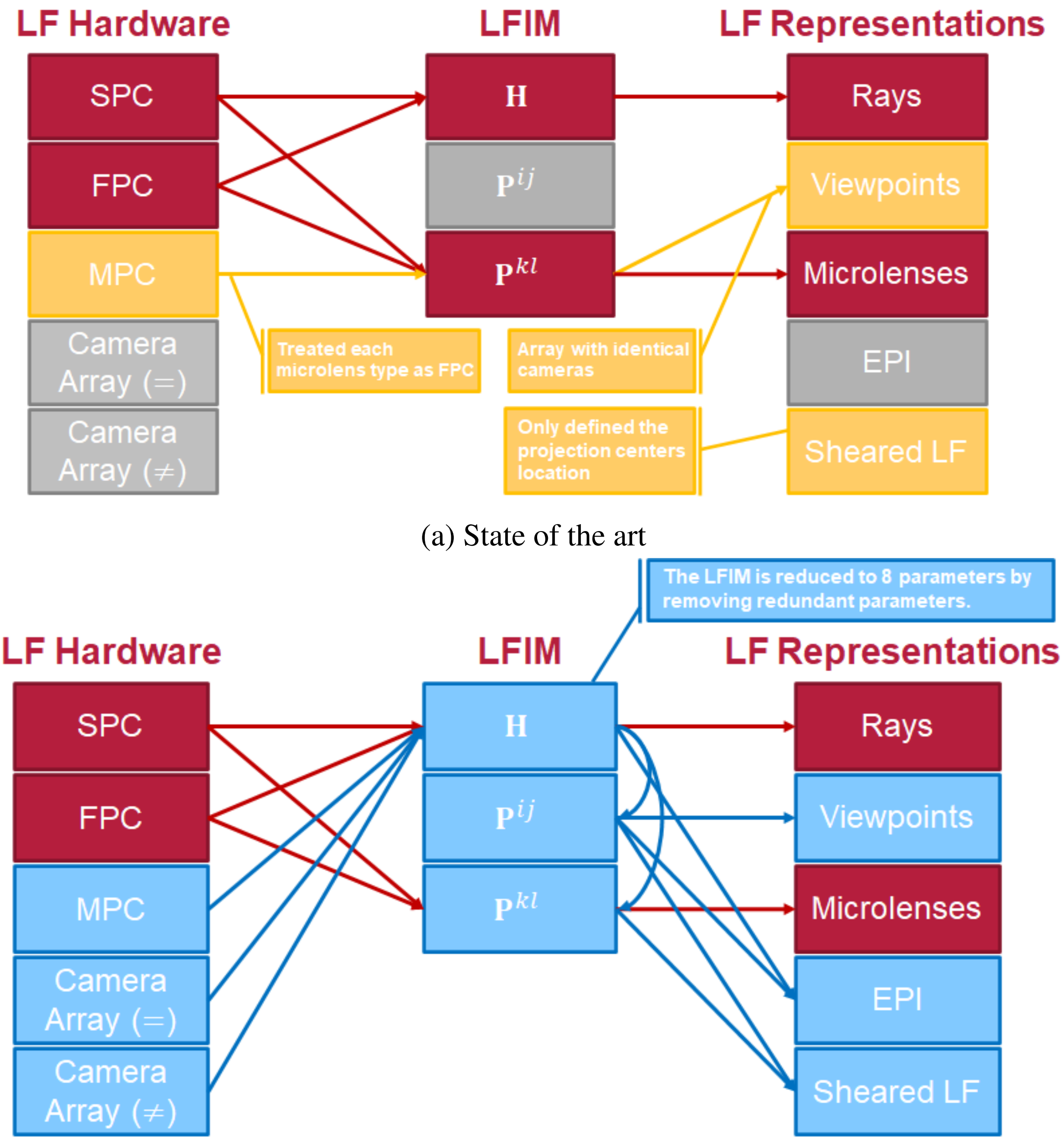
In the last decade, different plenoptic camera designs gave rise to various, specialized, geometric camera models [21, 41, 148]. The Lightfield Intrinsic Matrix (LFIM) mapping rays in the image space to rays in the object space is the most well known model to describe plenoptic cameras. This model appears in the literature with different structures, namely, Dansereau *et al.* [41] introduced this structure for Standard Plenoptic Cameras (SPCs) with 12 non-zero entries while Zhang *et al.* [150] presented the LFIM for SPCs and Focused Plenoptic Cameras (FPCs) with 6 non-zero entries. SPCs and FPCs can also be represented by describing the microlens cameras using a pinhole-like model with 6 parameters [21]. This model was extended to Multifocus Plenoptic Cameras (MPCs) by Nousias *et al.* [115] that considered each microlens type as an independent and separate FPC. Although the different camera models describe the same plenoptic camera types, no connection among the models is found in the literature (Figure 8.1.a).

In this thesis, were studied the different models under a common framework. More specifically, the models in the literature were reduced to a LFIM with 8 non-zero entries. This representation is obtained removing the redundant parameters with the extrinsic parameters and shifting the rays parameterization plane (Sections 4.2 and 5.2). The model proposed by Zhang *et al.* [150] corresponds to a LFIM with 8 free intrinsic parameters if one includes the 2 additional intrinsic parameters added in the radial distortion model [150] (Appendix A) relatively to the model of Brown [25]. On the other hand, the model proposed by Bok *et al.* [21] can be represented by a LFIM constraining the microlenses centers coordinates on the raw image to be regularly spaced (Section 5.4). The same LFIM can describe the chief-ray point projections of a world point in the different microlenses types for an MPC (Section 5.5).

Additionally, were explained the assumptions that allow to have an identical LFIM for SPCs and FPCs. More in detail, the same structure is obtained by the appropriate choice of the microlens coordinates and by assuming that the image sensor is aligned with the microlens array (Section 3.5).

The LFIM is not restricted as a camera model for plenoptic cameras, it can also be used to describe a coplanar camera array of identical cameras [95] (Appendix B) using a representation similar to the one from Zhang *et al.* [150]. In this case, the LFIM encloses information regarding the baseline and the intrinsic matrix used to define the cameras in the array, *i.e.* 6 non-zero parameters. The LFIM with 8 non-zero entries allows to describe a coplanar camera array of distinct cameras. In this case, the two additional parameters encode a principal point shift between the cameras in the array (Sections 4.1 and 5.1). The connection between the different plenoptic camera models [21, 41, 150] and between the LFIM and a camera array allow to switch representations regardless of the camera models considered during the calibration procedure (Figure 8.1.b).

The LFIM was also extended to model a camera array composed of identical coplanar plenoptic cameras with the same world focal plane. This setup corresponds to a multi-baseline camera array (Section 6.2).


LF Hardware
LFIM
LF Representations
SPC
FPC
MPC
Camera Array (=)
Camera Array (≠)
H
P^{ij}
P^{kl}
Rays
Viewpoints
Microlenses
EPI
Sheared LF
Treated each microlens type as FPC
Array with identical cameras
Only defined the projection centers location
The LFIM is reduced to 8 parameters by removing redundant parameters.

(a) State of the art

(b) Contributions

Figure 8.1: Relationships among camera models for plenoptic cameras and the types of images that can be obtained from an acquired LF. State of the art is summarized in **(a)** and the contributions are shown in **(b)**. The models and relationships in red are state of the art, in yellow are present in the literature but do not characterize completely the corresponding cameras, in grey are not found in the literature and in blue are denoted the contributions of the thesis.

8.2 Camera Array-based Representation for Plenoptic Cameras

The LF acquired by plenoptic cameras can be represented using Viewpoint Images (VIs) or Microlens Images (MIs) [112]. Although the arrays corresponding to these images have been considered in the literature [21, 112], their full projection model was yet to be formalized and there was not available a connection between these models and the LFIM [41] (Figure 8.1.a). A proposal of a mapping to the pinhole projection matrix [48] allows the LFIM to be adopted as mainstream in computer vision.

In this thesis, is defined the mapping among the LFIM and the virtual array (viewpoint camera array) and the physical array (microlens camera array) pinhole-type projection models. The viewpoint and microlens cameras define a coplanar array of cameras that differ on the location of their projection centers and on their principal points (Sections 4.1 and 5.1). The different principal points define an Epipolar Plane Image (EPI) geometry whose zero disparity plane is at a finite depth. This geometry extended the geometry defined by Bolles *et al.* [22] that considers images acquired by identical cameras, *i.e.* same principal point.

The rays captured by a plenoptic camera can be represented by a family of camera array models alternative to the viewpoint and microlens camera arrays [21, 104, 108]. In Chapter 6, is defined a constraint to obtain the set of rays in the image space that intersect at an arbitrary point in the object space. Using this contraint, one may extend the projection models associated with the viewpoint and microlens cameras to cope with the resampling of the LF which allows to define camera arrays at different depths. The camera arrays redefined for specific depths may differ only in their principal points comparing with the non-resampled camera arrays or may describe completely different arrays. More specifically, the resampling changes the location of the projection centers and the intrinsic parameters of the cameras in the array. The resampling of the LF also changes the EPI geometry and consequently the zero disparity plane depth. The contributions provided together with the contributions in Section 8.1 facilitate the use of plenoptic cameras by modeling them based on classic projection models.

The formalization of the microlens and viewpoint camera array representations, based on the pinhole camera model, enables novel calibration procedures for plenoptic cameras. More in detail, in Section 4.3, is proposed a calibration procedure for an SPC using corner points in VIs based on the viewpoint camera array projection model. The calibration consists in a linear solution capable of estimating the 8 parameters of the LFIM and a nonlinear optimization by minimizing the ray reprojection error. As in Zhang *et al.* [150], one only needs to estimate one homography for each calibration pattern pose, and extending techniques from pinhole camera calibration [151] to consider a coplanar camera array of distinct cameras, one is able to estimate the additional parameter regarding the baseline and principal point shift. To the best of our knowledge, this is the first work capable of estimating the principal point shift in the linear solution which allows to outperform state of the art methods.

In Section 5.5, one extended the LFIM used in SPCs and FPCs to MPCs. In this section, is proposed a camera model that describes the chief-ray point projections of a world point in the different microlenses using a single LFIM and the specific defocus behavior of each microlens type using the blur radius derived from the models [11, 18]. The microlens camera array projection model complemented by the blur model [11, 18] is used to define a calibration procedure for the MPC based on corner points and blur radius detected in the MIs using a new detection algorithm that overcomes the defocus blur present in the MIs. The corner detection algorithm and the calibration proposed outperform the state of the art showing that the MPC can be described using common intrinsic and extrinsic parameters among the different microlens types.

The extension of the LFIM to a coplanar array of plenoptic cameras allowed to describe the equivalent optical setup as a multi-baseline camera array: the baseline among the viewpoint cameras within a plenoptic camera and the baseline among plenoptic cameras. Using this equivalence, was proposed a calibration procedure for the plenoptic camera array that allowed to estimate all parameters in the linear solution (Section 6.2.2).

8.3 In Depth Study of Standard Plenoptic Cameras

The SPC was the first plenoptic camera available commercially. The low cost of this camera and the availability of an open source toolbox for the raw image decoding and processing [41] aided in the early adoption of this camera by the research community. Although this camera provides information regarding the depth of the scene, there was not found in the literature an evaluation of the depth capabilities of SPCs. Works evaluating the depth capabilities of plenoptic cameras were found just for FPCs. Hence, in Section 4.5, was studied the depth capabilities of an SPC in a range up to two meters considering several datasets with different zoom and focus settings captured by a 1^{st} generation Lytro camera. Experiments have shown that the depth error has a minimum at the main lens focal plane and increases as one moves away from this plane. Additionally, the increasing zoom allows decreasing the reconstruction error while the focus depth determines the depth range of the camera.

The SPC manufacturer provides information regarding the camera optical settings together with each acquired image. However, public domain calibration procedures for SPCs [21, 41] do not consider the information provided by the camera manufacturer and therefore rely completely on the acquisition of a dataset with a calibration pattern for a specific zoom and focus settings. In Section 4.4, are identified the relationships among the optical parameters provided as metadata as well as the relationships between these optical parameters and the entries of the viewpoint camera array projection model for different zoom and focus settings of the camera. These relationships allowed to define a regression model for the LFIM based on the observations of the main lens focal length and the infinity lambda meta-parameters provided by the manufacturer. This allowed to calibrate an SPC for a given zoom and focus settings without having to acquire a new calibration dataset.

8.4 Depth Estimation Boosting and Refinement

Plenoptic cameras acquire different perspectives of a point in a single acquisition. The multiple perspectives allow to recover the point's 3D information using several strategies ranging from stereo approaches to neural networks that explore the depth cues in the LF. In this thesis, are presented contributions to improve the multiview stereo, gradient-based, cost volume and deep neural network approaches.

In a multiview stereo approach, depth is estimated assuming no particular position for the microlens or viewpoint cameras by performing matching among the features detected on the corresponding images [3]. However, this strategy does not consider the regularity of the array of cameras defined by a plenoptic camera whose coplanarity and spacing set constraints on the projections observed on the multiple cameras (Sections 4.3 and 5.5). Consequently, is proposed a metric reconstruction methodology that ensures the projection geometry cues are satisfied (Section 7.2). The projection geometry cues proposed restrict the projections of a point to define lines in the ray-spaces (i, k) and (j, l). This approach improved the precision of the reconstruction by reducing the effects of discretization and independent detection of the features in each image.

The narrow baseline among cameras in the camera array equivalent representation of a plenoptic camera allows to apply gradient-based approaches in the EPIs [39, 139]. In this thesis, are reviewed these approaches to consider the EPI geometry of plenoptic cameras defined in Sections 4.1.3 and 5.1.3. In Section 7.3, is also derived the disparity estimation from LF gradients by using the concept of optical flow and the original concept of affine LF.

The affine LF concept is used in the context of cost-based approaches to gain insights on the metrics normally used (Section 7.4). Namely, the correspondence metric for an affine LF is a parabola with a minimum at the disparity value and with curvature that depends on the VI gradients. This expression allowed to define a methodology for disparity refinement and to identify that the curvature can be used as a confidence metric. Additionally, is shown the property that a focal stack of an affine LF does not contain disparity information. Depth has to be estimated directly from the affine LF.

The previous approaches allow reconstructing sparse disparity maps. In order to obtain dense disparity estimates it is necessary to use regularization methodologies. These approaches normally use just a subset of the LF information to recover disparity for the central view to reduce the high computational requirements [139]. In this thesis, is recovered a dense disparity map for all views by formalizing a data fusion problem which considers disparity estimates for each 2D EPI of the full 4D LF (Section 7.5). This approach is computed efficiently by resorting to an Alternating Direction Method of Multipliers (ADMM) instance, namely SALSA [4], and considering periodic boundary conditions that allowed to use the frequency domain.

In recent years, deep neural networks have been proposed to retrieve disparity maps from LFs [65, 129]. Neural networks generally require intense training, and while they may lead to good results, they may also result in an inability to perform well under inputs with characteristics outside their training scope. Normally, one can augment the training data [130] for the network to generalize for other inputs or one can perform transfer learning [47]. However, this is not always possible due to constraints such as lack of data, time or computational power.

In this thesis, is extended the application range of the LF depth estimation neural network [65]. The method extends the application range by transforming the input LF so that it falls under the training conditions. The proposed method provides a faster and more versatile approach at the cost of loosing some accuracy relatively to a full retraining approach.

8.5 Future Work

Plenoptic cameras acquire more information than conventional cameras. The additional information can be used to decompose the acquired LF into intrinsic components like albedo, shading and specularity [7, 8], or retrieve disparity or depth [139] that can be used to aid in other applications like image segmentation [141] and change detection [43]. Nonetheless, the amount of data generated to have this additional information and the characteristics of plenoptic cameras introduce some limitations detailed in the following for identifying possible points for future work.

Lower Cost and Wider Field of View (FOV) Optical Setups. The low-cost Lytro plenoptic cameras [113] stopped being available commercially in 2018. The plenoptic cameras available, nowadays, are expensive which limits their research and usage. It is important to design new and lower cost optical setups or pieces of hardware that allow conventional cameras to acquire the LF [13]. The equivalence of the plenoptic camera with a regular coplanar camera array may guide the setups in this direction (Sections 4.1 and 5.1). Nonetheless, solutions avoiding or reducing the problems identified in Section 2.3 for camera array setups are necessary.

The narrow baseline in plenoptic cameras allow to use gradient operators for easily recovering disparity estimates [39] when compared to a stereo-matching approach. However, this also results in a limited FOV that can be too restrictive for some applications. In a camera array design the baselines are inherently larger due to the encasing of the individual cameras which would result in a wider FOV. However, the larger baselines prevent the usage of gradient operators to retrieve disparity so one should come up with new solutions to synthesize intermediate views or to estimate disparity maps [110, 137]. An example of this setup can be for example the array of plenoptic cameras idealized in Section 6.2. Designs using combination of different cameras instead of similar cameras should also be thought.

At a first sight, another solution would be to replace the main lens by a wide angle lens. However, wide angle lenses are not easily adapted for LF acquisition [42]. Even so, the design of a single lens wide FOV LF camera is possible using wide FOV imaging techniques based on monocentric optics [42, 128].

Event-based LF Video. Plenoptic cameras acquire different perspectives of the scene which imply a high storage space demand. For example, the lowest resolution plenoptic camera acquires a 382×381 sub-aperture image with 11×11 directions per pixel which is equivalent to eight 1980×1080 pixels standard images. The area of LF compression is an active topic of research [36, 37, 99] and the Joint Photographic Experts Group (JPEG) is defining the standard JPEG Pleno that provides a standard framework to represent new imaging modalities including LF [9, 127]. The compression of LF will allow to reduce the requirements in terms of storage space and communications. However, since these compression methodologies work on the already acquired LF, the high throughput of the cameras is not reduced. In fact, current commercial plenoptic cameras providing LF video are rare and the frame rate at which they operate is still far from conventional cameras [120].

The recent event-based cameras [14, 87] mimic our retina [97] behavior by representing an image with spikes that report differences in image intensity. This allows a sparse representation of the image which allows to have higher acquisition rates. The event-based cameras sparse representation can be a solution to reduce the high throughput of plenoptic cameras and enable LF video within cost effective hardware solutions.

LF Video for 3D TV. The acquisition of LFs either requires a heavy post-processing if one considers images taken from arbitrary positions [57] or requires images taken from a fixed and regular geometry on a planar, cylindrical or spherical surface [84, 86]. The latter introduce specific geometries on the EPIs that help estimating disparity maps [22, 38]. In camera arrays, the precise placement and orientation of the individual cameras is hard and the process of converting the acquired images into a LF based on the two-plane parameterization implies loosing some of the acquired information. Hence, it is important to devise new LF representations that give more flexibility on the placement of the cameras [152].

The representations should also take into account domains of application. For example, the Surface Camera Image (SCam) captures the behavior of a surface point by characterizing the point from different views. This representation can be used to represent the surface of objects allowing to obtain an object-centered LF that is taylored for augmented reality. However, this approach can have high memory requirements according with the density of points used to represent the object.

The SCam representation is usually not the most convenient for display. Normally, the screens used to display LF information are planar and the scene has objects at multiple depths. The rays coming from the objects would have to be intersected on a plane for displaying purposes. This suggests that the two-plane parameterization representation for the LF is a better fit for displaying purposes and, consequently, for 3D TV [12, 146]. The challenge on these displays is more on the hardware side. At each point of the display it is necessary to differentiate the intensity being emitted according to the user's viewpoint. Additionally, in order to provide a continuous experience for the user it is necessary to obtain a continuous flow of viewpoints from the discretized LF.

Appendix A

Zhang *et al*. Mapping to LFIM

The model proposed by Zhang *et al.* [150] considers that the minimal form for the 5×5 Lightfield Intrinsic Matrix (LFIM) $\mathbf{H}_z$ has 6 non-zero entries

$$\mathbf{H}_z = \begin{bmatrix} h_{si} & 0 & 0 & 0 & 0 \\ 0 & h_{tj} & 0 & 0 & 0 \\ 0 & 0 & h_{uk} & 0 & h_u \\ 0 & 0 & 0 & h_{vl} & h_v \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} . \tag{A.1}$$

This model is complemented with a radial distortion model that has 2 additional parameters (k_3, k_4) relatively to [25]

$$\begin{cases} u^u = \alpha_r \, u + k_3 \, s \\ v^u = \alpha_r \, v + k_4 \, t \end{cases} , \tag{A.2}$$

where $\alpha_r = \left(1 + k_1 \, r^2 + k_2 \, r^4\right)$ corresponds to the standard radial distortion correction defined by Brown [25] with $r^2 = u^2 + v^2$, $[k_1, k_2, k_3, k_4]^T$ denotes the distortion vector, and $\boldsymbol{\Psi}^u = [s, t, u^u, v^u]^T$ is the undistorted ray in the object space.

A similar representation for the LFIM using 6 non-zero parameters has been proposed by Marto *et al.* [95] (Appendix B) to represent a camera array with coplanar projection centers and whose cameras are identical (same intrinsic parameters). Nonetheless, this is not the case for plenoptic cameras, *i.e.* the cameras in the array are not identical [104, 105] (Sections 4.1 and 5.1).

The LFIM (4.17) has a minimal form with 8 non-zero parameters as a consequence of the different intrinsic parameters between the viewpoint or microlens cameras. In this section, one shows that in fact the model proposed by Zhang *et al.* [150] is equivalent to the 8 non-zero parameters representation for the LFIM (4.17) considering that the two additional radial distortion parameters defined in Zhang *et al.* [150] relatively to Brown [25] are included in the $\mathbf{H}_z$ matrix. The two additional parameters for example, in the Standard Plenoptic Camera (SPC), are responsible for defining an Epipolar Plane Image (EPI) geometry that is consistent with the zero disparity plane at the main lens world focal plane [104, 107].

Let us define $u' = u + s\frac{k_3}{\alpha_r}$ and $v' = v + t\frac{k_4}{\alpha_r}$ to convert the Zhang *et al.* [150] radial distortion model (A.2) to the model defined by Brown [25] assuming that the α_r is constant. Considering the relationship between a point $[x, y, z]^T$ in the object space and the distorted ray $\mathbf{\Psi}_z = [s, t, u, v]^T$ in the object space as $[x, y]^T = [s, t]^T + z\ [u, v]^T$, and replacing the direction coordinates (u, v), one has $[x, y]^T = [s, t]^T + z[u' - s\frac{k_3}{\alpha_r}, v' - t\frac{k_4}{\alpha_r}]^T$. In order to obtain a relationship of the form $[x, y]^T = [s, t]^T + z[u', v']^T$, let us define the mapping between the rays in the object space $\mathbf{\Psi} = [s, t, u', v']^T$ and $\mathbf{\Psi}_z$ as

$$\tilde{\mathbf{\Psi}} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ \frac{k_3}{\alpha_r} & 0 & 1 & 0 & 0 \\ 0 & \frac{k_4}{\alpha_r} & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{H}^{\Psi}_{\Psi_z}} \tilde{\mathbf{\Psi}}_z \quad . \tag{A.3}$$

Extending the definition of $\mathbf{\Psi}_z$ to consider the rays in the image space

$\mathbf{\Phi} = [i, j, k, l]^T$ using the LFIM $\mathbf{H}_z$ proposed by Zhang *et al.* [150], one obtains a LFIM $\mathbf{H}'_z = \mathbf{H}^{\Psi}_{\Psi_z}\mathbf{H}_z$ defined with 8 non-zero entries

$$\mathbf{H}'_z = \begin{bmatrix} h_{si} & 0 & 0 & 0 & 0 \\ 0 & h_{tj} & 0 & 0 & 0 \\ h_{uk}\,\frac{k_3}{\alpha_r} & 0 & h_{uk} & 0 & h_u \\ 0 & h_{vl}\,\frac{k_4}{\alpha_r} & 0 & h_{vl} & h_v \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} . \tag{A.4}$$

Appendix B

Coplanar Camera Array of Identical Cameras

The Lightfield (LF) can be captured by camera arrays [144] (Section 2.3). The LFIM can represent an array comprised of several identical coplanar cameras by setting the principal point shift to zero, *i.e.* $\Delta u_0 = \Delta v_0 = 0$.

Let us consider the coordinates $(s, t, 0)$ that denote the position of the cameras' projection centers in a plane Γ, and $\left(u_p, v_p, f\right)$ that denote the points captured by a camera on a plane defined at a distance f from, and parallel to, the plane Γ. In this camera array setup, each image is obtained pointing in the same direction, and using identical cameras. Hence, nothing changes from camera to camera apart from their position (s, t), which does not affect the coordinates $\left(u_p, v_p\right)$ because of their local parameterization relatively to (s, t). Therefore, in this setup (s, t) is independent from $\left(u_p, v_p\right)$ and can be analyzed separately.

Regarding the (s, t) coordinates, one can assume that the projection centers of the cameras (i, j) are equally spaced in the plane Γ and the distance between consecutive projection centers is denoted by h_{si} and h_{tj}, which leads to

$$\begin{bmatrix} s \\ t \\ 1 \end{bmatrix} = \begin{bmatrix} h_{si} & 0 & 0 \\ 0 & h_{tj} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} i \\ j \\ 1 \end{bmatrix} \quad , \tag{B.1}$$

considering that $(s, t) = (0, 0)$ when $(i, j) = (0, 0)$.

Regarding the (u_p, v_p) coordinates, one can use the formula for the pinhole camera to describe the relationship between a world point and the pixel coordinates (k, l). This is, essentially, projecting the points on the plane which is f units of distance away from the plane Γ containing the array onto the image plane, *i.e.* $[k, l, 1]^T = \mathbf{K} \left[u_p, v_p, f\right]^T$.

Inverting the projection equation and combining with (B.1), the LFIM $\mathbf{H}$ is defined as

$$\mathbf{H} = \begin{bmatrix} h_{si} & 0 & \\ 0 & h_{tj} & \mathbf{0_{2\times 3}} \\ \mathbf{0_{3\times 2}} & & \mathbf{K}^{-1} \end{bmatrix} \quad \text{with} \quad \mathbf{K} = \begin{bmatrix} \frac{1}{h_{uk}} & 0 & -\frac{h_u}{h_{uk}} \\ 0 & \frac{1}{h_{vl}} & -\frac{h_v}{h_{vl}} \\ 0 & 0 & 1 \end{bmatrix} \tag{B.2}$$

where $\mathbf{0}_{n\times m}$ is the $n \times m$ null matrix, $\left[-h_{si}, -h_{tj}\right]^T$ corresponds to the baseline between consecutive cameras, and $\mathbf{K}$ corresponds to the intrinsic matrix that represents the cameras in the camera array defined using the LFIM (4.17) entries with $h_{ui} = h_{vj} = 0$. This reduces the EPI geometry (4.15) to the one presented by Bolles *et al.* [22]

$$\frac{\Delta k}{\Delta i} = k_u \frac{\Delta x_0}{z} \quad \text{and} \quad \frac{\Delta l}{\Delta j} = k_v \frac{\Delta y_0}{z} \,. \tag{B.3}$$

Appendix C

3D Corner-based Calibration

Let us consider the viewpoint projection matrix $\mathbf{P}^{ij}$ defined in equation (4.1) that maps a point $\mathbf{m} = [x, y, z]^T$ in the object space defined in the world coordinate system to a point in the image plane $\mathbf{q} = [k, l]^T$ through (4.19). The projection matrix entries of a viewpoint camera can be estimated from a set of tridimensional points, in the object space, and the corresponding image points using a Direct Linear Transformation (DLT) [63].

Let us consider that the viewpoint projection matrix $\mathbf{P}^{ij}$ associated with the viewpoint coordinates (i, j) can be defined from the projection matrix $\mathbf{P}^0$ associated with the viewpoint coordinates $(i, j) = (0, 0)$ and the projection viewpoint change matrix $\mathbf{A}^{ij}$ by

$$\mathbf{P}^{ij} = \underbrace{\begin{bmatrix} p^0_{11} & p^0_{12} & p^0_{13} & p^0_{14} \\ p^0_{21} & p^0_{22} & p^0_{23} & p^0_{24} \\ p^0_{31} & p^0_{32} & p^0_{33} & p^0_{34} \end{bmatrix}}_{\mathbf{P}^0} + \begin{bmatrix} i & 0 & 0 \\ 0 & j & 0 \\ 0 & 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{A}^{ij}} \quad , \qquad \text{(C.1)}$$

Considering the projection of a point $\mathbf{m}$ in the object space to the image point $\mathbf{q}$ for the viewpoint camera (i, j), applying the cross product by $\tilde{\mathbf{q}}$ on each side of the projection equation leads to $[\tilde{\mathbf{q}}]_\times \mathbf{P}^{ij} \tilde{\mathbf{m}} = \mathbf{0}_{3\times1}$, where $\left[(\cdot)\right]_\times$ is a skew-symmetric matrix that applies the cross product and $\mathbf{0}_{3\times1}$ is a 3×1 null matrix. Using the properties of the Kronecker product [93] and solving for each of the unknown parameters, one can redefine the projection equation as

$$\left(\tilde{\mathbf{m}}^T \otimes [\tilde{\mathbf{q}}]_\times\right) \mathbf{T} \begin{bmatrix} \mathbf{p}^0 \\ \mathbf{a}^{ij} \end{bmatrix} \tag{C.2}$$

with

$$\mathbf{T} = \left[\begin{array}{c} \mathbf{I}_{12\times 12} \end{array} \begin{array}{cccccccc} i & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & j & 0 & 0 & 0 & 0 & 0 & 0 \\ \multicolumn{8}{c}{\mathbf{0}_{1\times 8}} \\ 0 & 0 & i & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & j & 0 & 0 & 0 & 0 \\ \multicolumn{8}{c}{\mathbf{0}_{1\times 8}} \\ 0 & 0 & 0 & 0 & i & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & j & 0 & 0 \\ \multicolumn{8}{c}{\mathbf{0}_{1\times 8}} \\ 0 & 0 & 0 & 0 & 0 & 0 & i & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & j \\ \multicolumn{8}{c}{\mathbf{0}_{1\times 8}} \end{array} \right] , \tag{C.3}$$

where $\mathbf{p}^0$ and $\mathbf{a}^{ij}$ correspond to vectorizations of the matrix $\mathbf{P}^0$ and $\mathbf{A}^{ij}$ by stacking their columns, respectively. $\mathbf{I}_{12\times 12}$ is a 12×12 identity matrix and $\mathbf{0}_{1\times 8}$ is a 1×8 null matrix. The solution $\left[\mathbf{p}^0, \mathbf{a}^{ij}\right]^T$ for the parametric projection matrix can be estimated using Singular Value Decomposition (SVD).

The restrictions introduced by a plenoptic camera allow to represent the parametric projection matrix (C.1) using 20 parameters. According to equation (C.2), each pair $(\tilde{\mathbf{m}}, \tilde{\mathbf{q}})$ originates three equations with only two being linearly independent. On the other hand, each point in the object space originates N image points, one for each viewpoint camera, assuming that the point is observed in all viewpoint cameras. However, as discussed in Section 4.3.1, the number of linearly independent equations originated by a point in the object space is 4. Thus, one needs at least 5 non-coplanar points in the object space to obtain the entries of the projection matrix $\mathbf{P}^{ij}$.

Appendix D

Reconstruction Estimation Accuracy without Distortion

The results of the reconstruction estimation accuracy not considering radial distortion correction are summarized in Table D.1. Comparing with Table 4.11, one can see that the results are very similar which allows to conclude that the radial distortion does not play an important role on the reconstruction estimation accuracy. Nonetheless, one has to notice that the depth range with normalized reconstruction error lower or equal to 10% is larger for Datasets E and F, and that Dataset C has a normalized reconstruction error that is always greater than the 10% for all depth range analyzed.

Dataset	**Depth Range (m)**	**Mean ± STD Error in Depth Range (%)**	**Mean ± STD Error (%)**
A	0.35 - 1.30	6.84 ± 5.11	16.67 ± 6.28
B	0.40 - 1.30	7.89 ± 5.96	13.72 ± 9.73
C	Not Defined	Not Defined	23.74 ± 17.72
D	0.60 - 2.00	5.18 ± 3.14	14.18 ± 4.87
E	0.65 - 2.00	5.48 ± 3.04	8.05 ± 4.01
F	0.85 - 2.00	3.97 ± 1.42	5.67 ± 1.57
G	1.50 - 1.85	1.94 ± 0.61	1.94 ± 0.61

Table D.1: Depth ranges for the datasets acquired not considering radial distortion correction. The depth ranges are identified as the regions whose mean for the normalized reconstruction errors is lower or equal to 10%. The mean and STD for the normalized reconstruction errors within the depth ranges defined and for all ground truth depths are also depicted.

Appendix E

Microlens Projection Centers

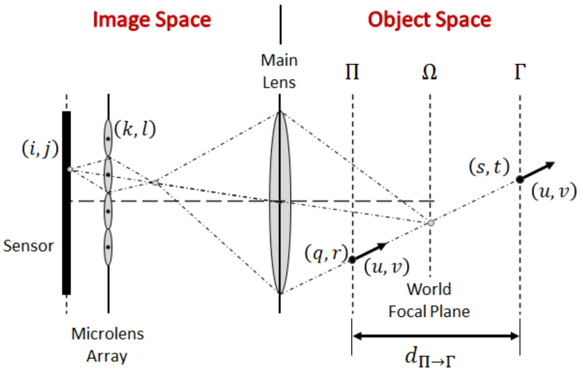


Figure E.1: LF parameterization. The LF in the image space is parameterized using pixels and microlenses indices while the LF in the object space is parameterized using a point and a direction.

Let us consider the LF in the object space $L_{\Pi}\left(q, r, u, v\right)$ where each ray $\mathbf{\Psi}_{\Pi} = \left[q, r, u, v\right]^T$ is parameterized using a point $\left[q, r, 0\right]^T$ on a plane Π and a direction $\left[u, v, 1\right]^T$ defined in metric units [107] (Figure E.1). The LFIM $\mathbf{H}_{\Pi}$ (4.7) maps this LF to the LF in the image space $L\left(i, j, k, l\right)$ by (4.6) where $\mathbf{\Phi} = \left[i, j, k, l\right]^T$ corresponds to a ray that is parameterized by pixels (i, j) and microlenses (k, l) indices. For a microlens camera, the microlens coordinates (k, l) are fixed and are considered as parameters. Hence, for a microlens camera, the positions (q, r) and the directions (u, v) are affine mappings only on the pixel coordinates (i, j), namely

$$\begin{cases} q\left(i;\ k, \mathbf{H}_{\Pi}\right) = h_{qi}\, i + h_{qk}\, k + h_q \\ r\left(j;\ l, \mathbf{H}_{\Pi}\right) = h_{rj}\, j + h_{rl}\, l + h_r \\ u\left(i;\ k, \mathbf{H}_{\Pi}\right) = h_{ui}\, i + h_{uk}\, k + h_u \\ v\left(j;\ l, \mathbf{H}_{\Pi}\right) = h_{vj}\, j + h_{vl}\, l + h_v \end{cases} \tag{E.1}$$

where the LFIM $\mathbf{H}_{\Pi}$ is also considered as a parameter. To simplify the notation, the parameters $(k, l, \mathbf{H}_{\Pi})$ will not be included in the following expressions.

A ray captured by a plenoptic camera and parameterized by (i, j, k, l) intersects the plane Π at point $\mathbf{p}\left(i, j\right) = [q(i),\ r(j),\ 0]^T$ with a direction $\mathbf{n}\left(i, j\right) = [u(i),\ v(j),\ 1]^T$. This allows to define an arbitrary point $\mathbf{c}\left(i, j, \lambda\right) = [x, y, z]^T$ along the ray [58] as

$$\mathbf{c}\left(i, j, \lambda\right) = \mathbf{p}\left(i, j\right) + \lambda\, \mathbf{n}\left(i, j\right) \ ,\ \lambda \in \mathbb{R} \quad . \tag{E.2}$$

Note that by sweeping the range of (i, j) in (E.2) with $\lambda = 0$, one samples an area of the plane Π through which pass all the microlens imaging rays. In addition, by sweeping (k, l), one obtains all the microlens cameras, and therefore all rays that can be imaged by the plenoptic camera. Finally, sweeping λ, allows representing all world points within the Field of View (FOV) of the plenoptic camera.

The location of the projection centers of an optical setup is defined by its caustic surface, which is the loci of singularities in the flux density [27, 58]. The convergence of the rays captured by a camera at a single point, *i.e.* a unique projection center, is considered a degenerate configuration of the caustic surface (point caustic) [58]. Although there are many techniques to derive the caustic surface, one will consider the Jacobian method [27].

The caustic surface is defined at the points in the object space where the ray to image mapping (E.2) is singular, *i.e.* the mapping from (i, j, λ) to (x, y, z) is singular. The singularities occur at the set of points where the Jacobian matrix of the transformation does not have full rank, *i.e.* points that make the determinant of the Jacobian vanish $\det\Big(\mathbf{J}\left(\mathbf{c}\left(i,j,\lambda\right)\right)\Big) = 0$. Solving the vanishing constraint one obtains two solutions for λ:

$$\lambda_1 = -\frac{h_{qi}}{h_{ui}} \quad \vee \quad \lambda_2 = -\frac{h_{rj}}{h_{vj}} \quad . \tag{E.3}$$

Replacing λ_1 or λ_2 in (E.2), one identifies the caustic profile for the microlens camera. The caustic profile of a single microlens consists of a line with (i) unique (x, z) and variable y components if $\lambda = \lambda_1$ or (ii) unique (y, z) and variable x components if $\lambda = \lambda_2$. In case $\lambda_1 \neq \lambda_2$ the microlens is a non-central camera. The microlens camera corresponds to a central camera, *i.e.* a camera with a unique projection center, if and only if $\lambda_1 = \lambda_2$ which imply the model parameters relation

$$\frac{h_{qi}}{h_{ui}} = \frac{h_{rj}}{h_{vj}} \quad . \tag{E.4}$$

Assuming this constraint and replacing λ in (E.2), expanded by the expressions in (E.1), the location of the microlens projection center for a microlens camera (k, l) is given by

$$\mathbf{p}_c = \begin{bmatrix} h_q - \frac{h_{qi}}{h_{ui}} h_u + k\left(h_{qk} - \frac{h_{qi}}{h_{ui}} h_{uk}\right) \\ h_r - \frac{h_{rj}}{h_{vj}} h_v + l\left(h_{rl} - \frac{h_{rj}}{h_{vj}} h_{vl}\right) \\ -\frac{h_{qi}}{h_{ui}} \end{bmatrix} \quad . \tag{E.5}$$

Furthermore, considering all microlens cameras that can be defined, the LFIM represents a coplanar grid of equally spaced projection centers. Notice that the microlens coordinates (k, l) only affect the x- and y-components of the projection centers while the z-component of the projections centers is always the same.

Appendix F

Disparity Estimation Data Fusion

In the use of the Alternating Direction Method of Multipliers (ADMM) instance, Split Augmented Lagrangian Shrinkage Algorithm (SALSA) [4], one is minimizing the augmented Lagrangian relatively to the auxiliary variables. So, let us define each of the minimization problems in the Algorithm 3. Expanding the equation (7.53), one has

$$\begin{aligned}\mathcal{L}\left(\mathbf{Z}, \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3, \mathbf{V}_4, \mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3, \mathbf{A}_4\right) = & \frac{1}{2}\left\|\mathbf{Y}_j - \mathbf{M}_j \mathbf{V}_1\right\|_F^2 \\ & + \frac{\lambda_i}{2}\left\|\mathbf{Y}_i^T - \mathbf{V}_2 \mathbf{M}_i\right\|_F^2 \\ & + \lambda_r \operatorname{TV}\left(\mathbf{V}_3, \mathbf{V}_4\right) + \\ & + \frac{\mu}{2}\left\|\mathbf{Z} - \mathbf{V}_1 - \mathbf{A}_1\right\|_F^2 \\ & + \frac{\mu}{2}\left\|\mathbf{Z} - \mathbf{V}_2 - \mathbf{A}_2\right\|_F^2 + \\ & + \frac{\mu}{2}\left\|\mathbf{Z}\mathbf{D}_h - \mathbf{V}_3 - \mathbf{A}_3\right\|_F^2 \\ & + \frac{\mu}{2}\left\|\mathbf{Z}\mathbf{D}_v - \mathbf{V}_4 - \mathbf{A}_4\right\|_F^2\end{aligned} \quad . \tag{F.1}$$

In the Algorithm 3, the first minimization problem is

$$\mathbf{Z}^{(k+1)} \in \arg \min_{\mathbf{Z}} \left(\frac{\mu}{2} \left\| \mathbf{Z} - \mathbf{V}_1^{(k)} - \mathbf{A}_1^{(k)} \right\|_F^2 + \frac{\mu}{2} \left\| \mathbf{Z} - \mathbf{V}_2^{(k)} - \mathbf{A}_2^{(k)} \right\|_F^2 + \right. \\ \left. + \frac{\mu}{2} \left\| \mathbf{Z}\mathbf{D}_h - \mathbf{V}_3^{(k)} - \mathbf{A}_3^{(k)} \right\|_F^2 + \frac{\mu}{2} \left\| \mathbf{Z}\mathbf{D}_v - \mathbf{V}_4^{(k)} - \mathbf{A}_4^{(k)} \right\|_F^2 \right) \tag{F.2}$$

which corresponds to a quadratic problem that has the following solution

$$\mathbf{Z}^{(k+1)} = \left[2\mathbf{I} + \mathbf{D}_h\mathbf{D}_h^T + \mathbf{D}_v\mathbf{D}_v^T \right]^{-1} \\ \left[\left(\mathbf{V}_1^{(k)} + \mathbf{A}_1^{(k)} \right) + \left(\mathbf{V}_2^{(k)} + \mathbf{A}_2^{(k)} \right) \right. \\ \left. + \left(\mathbf{V}_3^{(k)} + \mathbf{A}_3^{(k)} \right) \mathbf{D}_h^T + \left(\mathbf{V}_4^{(k)} + \mathbf{A}_4^{(k)} \right) \mathbf{D}_v^T \right] . \tag{F.3}$$

The inverse does not depend on the iteration and can be computed before entering the loop defined in Algorithm 3. The inverse can be computed using the Fast Fourier Transform (FFT) due to the periodic boundary conditions assumed.

The second minimization problem in Algorithm 3 is subdivided into three minimization problems. Let us consider the minimization problem for $\mathbf{V}_1$ defined as

$$\mathbf{V}_1^{(k+1)} \in \arg \min_{\mathbf{V}_1} \left(\frac{1}{2} \left\| \mathbf{Y}_j - \mathbf{M}_j \, \mathbf{V}_1 \right\|_F^2 + \frac{\mu}{2} \left\| \mathbf{Z}^{(k+1)} - \mathbf{V}_1 - \mathbf{A}_1^{(k)} \right\|_F^2 \right) . \tag{F.4}$$

This is also a quadratic problem. In order to solve this problem, let us split the variable $\mathbf{V}_1$ using the sampling matrix $\mathbf{M}_j$ into $\mathbf{M}_j\mathbf{V}_1$ and $\bar{\mathbf{M}}_j\mathbf{V}_1$. $\bar{\mathbf{M}}_j$ is the matrix that allows to select the pixels not selected by

$\mathbf{M}_j$. The solution to the minimization problem is given by

$$\begin{aligned} \mathbf{M}_j\mathbf{V}_1^{(k+1)} &= \frac{1}{1+\mu}\mathbf{M}_j\left[\mathbf{Y}_j + \mu\left(\mathbf{Z}^{(k+1)} - \mathbf{A}_1^{(k)}\right)\right] \\ \bar{\mathbf{M}}_j\mathbf{V}_1^{(k+1)} &= \bar{\mathbf{M}}_j\left(\mathbf{Z}^{(k+1)} - \mathbf{A}_1^{(k)}\right) \end{aligned} \quad . \tag{F.5}$$

The minimization problem for $\mathbf{V}_2$ is given by:

$$\mathbf{V}_2^{(k+1)} \in \arg\min_{\mathbf{V}_2} \left(\frac{\lambda_i}{2}\left\|\mathbf{Y}_i^T - \mathbf{V}_2\,\mathbf{M}_i\right\|_F^2 + \frac{\mu}{2}\left\|\mathbf{Z}^{(k+1)} - \mathbf{V}_2 - \mathbf{A}_2^{(k)}\right\|_F^2\right) \quad . \tag{F.6}$$

This minimization problem yields a similar solution to the minimization problem (F.4) with some minor changes due to the regularization parameter and the sampling matrix being different

$$\begin{aligned} \mathbf{V}_2^{(k+1)}\mathbf{M}_i &= \frac{1}{\lambda_i+\mu}\left[\lambda_i\mathbf{Y}_i^T + \mu\left(\mathbf{Z}^{(k+1)} - \mathbf{A}_2^{(k)}\right)\right]\mathbf{M}_i \\ \mathbf{V}_2^{(k+1)}\bar{\mathbf{M}}_i &= \left(\mathbf{Z}^{(k+1)} - \mathbf{A}_2^{(k)}\right)\bar{\mathbf{M}}_i \end{aligned} \quad . \tag{F.7}$$

The minimization problem with respect to $\mathbf{V}_3$ and $\mathbf{V}_4$ is:

$$\begin{aligned} \left\{\mathbf{V}_3^{(k+1)}, \mathbf{V}_4^{(k+1)}\right\} \in \arg\min_{\mathbf{V}_3,\mathbf{V}_4} \Bigg(&\lambda_r \mathrm{TV}\left(\mathbf{V}_3, \mathbf{V}_4\right) \\ &+ \frac{\mu}{2}\left\|\mathbf{Z}^{(k+1)}\mathbf{D}_h - \mathbf{V}_3 - \mathbf{A}_3^{(k)}\right\|_F^2 \\ &+ \frac{\mu}{2}\left\|\mathbf{Z}^{(k+1)}\mathbf{D}_v - \mathbf{V}_4 - \mathbf{A}_4^{(k)}\right\|_F^2\Bigg) \end{aligned} \tag{F.8}$$

which has the solution

$$\left\{\left(\mathbf{V}_3^{(k+1)}\right)_{:m},\left(\mathbf{V}_4^{(k+1)}\right)_{:m}\right\}=\max\left\{\|\mathbf{C}\|_1-\frac{\lambda_r}{\mu},0\right\}\ \mathrm{sign}\left(\mathbf{C}\right) \quad \text{(F.9)}$$

where $(\cdot)_{:m}$ denotes the m-th column of the matrix $(\cdot)$, $\|\cdot\|_1=\sum_i|\cdot|_i$ is the vector L1-norm, and

$$\mathbf{C}=\left\{\left(\mathbf{Z}^{(k+1)}\mathbf{D}_h-\mathbf{A}_3^{(k)}\right)_{:m},\left(\mathbf{Z}^{(k+1)}\mathbf{D}_v-\mathbf{A}_4^{(k)}\right)_{:m}\right\}. \quad \text{(F.10)}$$

This corresponds to the columnwise soft-threshold function. This minimization problem can be efficiently computed using FFTs.

Finally, one has to update the Lagrange multipliers $\mathbf{A}$:

$$\begin{aligned}\mathbf{A}_1^{(k+1)}&=\mathbf{A}_1^{(k)}-\left(\mathbf{Z}^{(k+1)}-\mathbf{V}_1^{(k+1)}\right)\\ \mathbf{A}_2^{(k+1)}&=\mathbf{A}_2^{(k)}-\left(\mathbf{Z}^{(k+1)}-\mathbf{V}_2^{(k+1)}\right)\\ \mathbf{A}_3^{(k+1)}&=\mathbf{A}_3^{(k)}-\left(\mathbf{Z}^{(k+1)}\mathbf{D}_h-\mathbf{V}_3^{(k+1)}\right)\\ \mathbf{A}_4^{(k+1)}&=\mathbf{A}_4^{(k)}-\left(\mathbf{Z}^{(k+1)}\mathbf{D}_v-\mathbf{V}_4^{(k+1)}\right)\end{aligned} \quad \text{(F.11)}$$

Bibliography

[1] Y. Abdel-Aziz. Karara. hm 1971. direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry. In *Proceedings ASP/VI Symp. On Close-Range Photogrammetry*, pages 1–17, 1971.

[2] E. H. Adelson and J. R. Bergen. *The plenoptic function and the elements of early vision.* Vision and Modeling Group, Media Laboratory, Massachusetts Institute of Technology, 1991.

[3] E. H. Adelson and J. Y. A. Wang. Single lens stereo with a plenoptic camera. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 14(2):99–106, 1992.

[4] M. V. Afonso, J. M. Bioucas-Dias, and M. A. Figueiredo. An augmented lagrangian approach to the constrained optimization formulation of imaging inverse problems. *Image Processing, IEEE Transactions on*, 20(3):681–695, 2011.

[5] W. Ahmad, L. Palmieri, R. Koch, and M. Sjöström. Matching light field datasets from plenoptic cameras 1.0 and 2.0. In *2018-3DTV-Conference: The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON)*, pages 1–4. IEEE, 2018.

[6] J. Allman and E. McGuinness. Visual cortex in primates. *Comparative primate biology*, 4:279–326, 1988.

[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)*, pages 2165–2169. IEEE, 2018.

[8] A. Alperovich, O. Johannsen, M. Strecke, and B. Goldluecke. Light field intrinsics with a deep encoder-decoder network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9145–9154, 2018.

[9] G. D. O. Alves, M. B. De Carvalho, C. L. Pagliari, P. G. Freitas, I. Seidel, M. P. Pereira, C. F. S. Vieira, V. Testoni, F. Pereira, and E. A. Da Silva. The jpeg pleno light field coding standard 4d-transform mode: How to design an efficient 4d-native codec. *IEEE Access*, 8:170807–170829, 2020.

[10] S. Arseneau and J. R. Cooperstock. An improved representation of junctions through asymmetric tensor diffusion. In *Advances in Visual Computing*, pages 363–372. Springer, 2006.

[11] M. Baba, M. Mukunoki, and N. Asada. A unified camera calibration using geometry and blur of feature points. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 1, pages 816–819. IEEE, 2006.

[12] T. Balogh, P. T. Kovács, and A. Barsi. Holovizio 3d display system. In *2007 3DTV Conference*, pages 1–4. IEEE, 2007.

[13] S. Bazeille, Y. Maillot, F. Cordier, C. Riou, and C. Cudel. Light-field image acquisition from a conventional camera: design of a four minilens ring device. *Optical Engineering*, 58(1):015105, 2019.

[14] R. Berner and T. Delbruck. Event-based pixel sensitive to changes of color and brightness. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 58(7):1581–1590, 2011.

[15] M. Bettini. Apiaria universae philosophiae mathematicae, 1642.

[16] J. Bigun. Optimal orientation detection of linear symmetry. 1987.

[17] C. Birklbauer and O. Bimber. Panorama light-field imaging. *Computer Graphics Forum*, 33(2):43–52, 2014.

[18] T. E. Bishop and P. Favaro. The light field camera: Extended depth of field, aliasing, and superresolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(5):972–986, 2012.

[19] K. Bitsakos and C. Fermüller. Depth estimation using the compound eye of dipteran flies. *Biological cybernetics*, 95(5):487–501, 2006.

[20] Y. Bok, H. Ha, and I. S. Kweon. Automated checkerboard detection and indexing using circular boundaries. *Pattern Recognition Letters*, 71:66–72, 2016.

[21] Y. Bok, H.-G. Jeon, and I. S. Kweon. Geometric calibration of micro-lens-based light field cameras using line features. *IEEE transactions on pattern analysis and machine intelligence*, 39(2):287–300, 2017.

[22] R. C. Bolles, H. H. Baker, and D. H. Marimont. Epipolar-plane image analysis: An approach to determining structure from motion. *International Journal of Computer Vision*, 1(1):7–55, 1987.

[23] A. Borst. Drosophila's view on insect vision. *Current biology*, 19(1):R36–R47, 2009.

[24] G. J. Brelstaff, A. Párraga, T. Troscianko, and D. Carr. Hyperspectral camera system: acquisition and analysis. In *Geographic Information Systems, Photogrammetry, and Geological/Geophysical Remote Sensing*, volume 2587, pages 150–159. International Society for Optics and Photonics, 1995.

[25] D. C. Brown. Decentering distortion of lenses. *Photogrammetric Engineering and Remote Sensing*, 1966.

[26] T. Brox, J. Weickert, B. Burgeth, and P. Mrázek. Nonlinear structure tensors. *Image and Vision Computing*, 24(1):41–55, 2006.

[27] D. G. Burkhard and D. L. Shealy. Flux density for ray propagation in geometrical optics. *JOSA*, 63(3):299–304, 1973.

[28] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.

[29] C. Chen, H. Lin, Z. Yu, S. Bing Kang, and J. Yu. Light field stereo matching using bilateral statistics of surface cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1518–1525, 2014.

[30] W.-C. Chen, J.-Y. Bouguet, M. H. Chu, and R. Grzeszczuk. Light field mapping: Efficient representation and hardware rendering of surface light fields. In *ACM Transactions on Graphics (TOG)*, volume 21, pages 447–456. ACM, 2002.

[31] W. Chi, K. Chu, and N. George. Polarization coded aperture. *Optics express*, 14(15):6634–6642, 2006.

[32] 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 *Proceedings of the IEEE International Conference on Computer Vision*, pages 3280–3287, 2013.

[33] T. R. Clandinin and S. L. Zipursky. Making connections in the fly visual system. *Neuron*, 35(5):827–841, 2002.

[34] L. Condat. A primal–dual splitting method for convex optimization involving lipschitzian, proximable and linear composite terms. *Journal of Optimization Theory and Applications*, 158(2):460–479, 2013.

[35] A. R. Conn, N. I. Gould, and P. L. Toint. *Trust region methods*, volume 1. Siam, 2000.

[36] C. Conti, P. Nunes, and L. D. Soares. Light field image coding with jointly estimated self-similarity bi-prediction. *Signal Processing: Image Communication*, 60:144–159, 2018.

[37] C. Conti, L. D. Soares, and P. Nunes. Dense light field coding: A survey. *IEEE Access*, 8:49244–49284, 2020.

[38] A. Cserkaszky, P. A. Kara, A. Barsi, M. G. Martini, and T. Balogh. Light-fields of circular camera arrays. In *2018 26th European Signal Processing Conference (EUSIPCO)*, pages 241–245. IEEE, 2018.

[39] D. Dansereau and L. Bruton. Gradient-based depth estimation from 4d light fields. In *Circuits and Systems, 2004. ISCAS'04. Proceedings of the 2004 International Symposium on*, volume 3, pages III–549. IEEE, 2004.

[40] D. G. Dansereau, B. Girod, and G. Wetzstein. Liff: Light field features in scale and depth. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8042–8051, 2019.

[41] D. G. Dansereau, O. Pizarro, and S. B. Williams. Decoding, calibration and rectification for lenselet-based plenoptic cameras. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1027–1034, 2013.

[42] D. G. Dansereau, G. Schuster, J. Ford, and G. Wetzstein. A wide-field-of-view monocentric light field camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5048–5057, 2017.

[43] D. G. Dansereau, S. B. Williams, and P. I. Corke. Simple change detection from mobile light field cameras. *Computer Vision and Image Understanding*, 145:160–171, 2016.

[44] P. David, M. Le Pendu, and C. Guillemot. White lenslet image guided demosaicing for plenoptic cameras. In *19th International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–6. IEEE, 2017.

[45] T. Delbrück and C. Mead. Analog vlsi phototransduction by continuous-time, adaptive, logarithmic photoreceptor circuits. 1995.

[46] M. Diebold and B. Goldluecke. Epipolar Plane Image Refocusing for Improved Depth Estimation and Occlusion Handling. In M. Bronstein, J. Favre, and K. Hormann, editors, *Vision, Modeling & Visualization*. The Eurographics Association, 2013.

[47] T. G. Dietterich, L. Pratt, and S. Thrun. Special issue on inductive transfer. *Machine Learning*, 28(1), 1997.

[48] O. Faugeras. *Three-dimensional computer vision: a geometric viewpoint*. MIT press, 1993.

[49] R. A. Fisher. Statistical methods for research workers. In *Breakthroughs in statistics*, pages 66–70. Springer, 1992.

[50] L. Gabriel. La photographie intégrale. *Comptes-Rendus, Académie des Sciences*, 146:446–551, 1908.

[51] J. E. Gentle. *Numerical linear algebra for applications in statistics*. Springer Science & Business Media, 2012.

[52] T. Georgiev and A. Lumsdaine. Reducing plenoptic camera artifacts. In *Computer Graphics Forum*, volume 29, pages 1955–1968. Wiley Online Library, 2010.

[53] T. Georgiev, A. Lumsdaine, and G. Chunev. Using focused plenoptic cameras for rich image capture. *IEEE Computer Graphics and Applications*, 31(1):62–73, 2010.

[54] T. Georgiev, K. C. Zheng, B. Curless, D. Salesin, S. Nayar, and C. Intwala. Spatio-angular resolution tradeoffs in integral photography. *Rendering Techniques*, 2006:263–272, 2006.

[55] A. Gershun. The light field. *Journal of Mathematics and Physics*, 18(1-4):51–151, 1939.

[56] B. Goldluecke and S. Wanner. The variational structure of disparity and regularization of 4d light fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1003–1010, 2013.

[57] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen. The lumigraph. In *Proceedings of the International conference on computer graphics and interactive techniques (SIGGRAPH)*, volume 96, pages 43–54. ACM, 1996.

[58] M. D. Grossberg and S. K. Nayar. The raxel imaging model and ray-based calibration. *International Journal of Computer Vision*, 61(2):119–137, 2005.

[59] C. Hahne, A. Aggoun, S. Haxha, V. Velisavljevic, and J. C. J. Fernández. Light field geometry of a standard plenoptic camera. *Optics express*, 22(22):26659–26673, 2014.

[60] C. Hahne, A. Aggoun, and V. Velisavljevic. The refocusing distance of a standard plenoptic photograph. In *2015 3DTV-Conference: The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON)*, pages 1–4. IEEE, 2015.

[61] C. Hahne, A. Aggoun, V. Velisavljevic, S. Fiebig, and M. Pesch. Refocusing distance of a standard plenoptic camera. *Optics Express*, 24(19):21521–21540, 2016.

[62] C. G. Harris, M. Stephens, et al. A combined corner and edge detector. In *Alvey vision conference*, volume 15, pages 10–5244. Citeseer, 1988.

[63] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[64] R. I. Hartley. In defence of the 8-point algorithm. In *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pages 1064–1070. IEEE, 1995.

[65] C. Hazirbas, L. Leal-Taixé, and D. Cremers. Deep depth from focus. *arXiv preprint arXiv:1704.01085*, 2017.

[66] C. Heinze, S. Spyropoulos, S. Hussmann, and C. Perwass. Automated robust metric calibration algorithm for multifocus plenoptic cameras. *IEEE Transactions on Instrumentation and Measurement*, 65(5):1197–1205, 2016.

[67] I. Her. Geometric transformations on the hexagonal grid. *IEEE Transactions on Image Processing*, 4(9):1213–1222, 1995.

[68] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldluecke. A dataset and evaluation methodology for depth estimation on 4d light fields. In *Asian Conference on Computer Vision*, pages 19–34. Springer, 2016.

[69] B. K. Horn and B. G. Schunck. Determining optical flow. In *Techniques and Applications of Image Understanding*, volume 281, pages 319–331. International Society for Optics and Photonics, 1981.

[70] I. P. Howard, B. J. Rogers, et al. *Binocular vision and stereopsis*. Oxford University Press, USA, 1995.

[71] I. Ihrke, J. Restrepo, and L. Mignard-Debise. Principles of light field imaging: Briefly revisiting 25 years of research. *IEEE Signal Processing Magazine*, 33(5):59–69, 2016.

[72] I. Ihrke, T. Stich, H. Gottschlich, M. Magnor, and H.-P. Seidel. Fast incident light field acquisition and rendering. *WSCG*, 16(1-3):25–32, 2008.

[73] A. Isaksen, L. McMillan, and S. J. Gortler. Dynamically reparameterized light fields. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 297–306. ACM Press/Addison-Wesley Publishing Co., 2000.

[74] H. E. Ives. A camera for making parallax panoramagrams. *JOSA*, 17(6):435–439, 1928.

[75] H. E. Ives. Parallax panoramagrams made with a large diameter lens. *JOSA*, 20(6):332–342, 1930.

[76] H.-G. Jeon, J. Park, G. Choe, J. Park, Y. Bok, Y.-W. Tai, and I. So Kweon. Accurate depth map estimation from a lenslet light field camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1547–1555, 2015.

[77] O. Johannsen, C. Heinze, B. Goldluecke, and C. Perwaß. On the calibration of focused plenoptic cameras. In *Time-of-Flight and Depth Imaging*, pages 302–317. Springer, 2013.

[78] A. Kassir and T. Peynot. Reliable automatic camera-laser calibration. In *Proceedings of the 2010 Australasian Conference on Robotics & Automation*. ARAA, 2010.

[79] D. G. Kendall. A survey of the statistical theory of shape. *Statistical Science*, pages 87–99, 1989.

[80] C. S. Kenney, M. Zuliani, and B. Manjunath. An axiomatic approach to corner detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 191–197. IEEE, 2005.

[81] K. Kirschfeld. The projection of the optical environment on the screen of the rhabdomere in the compound eye of the musca. *Experimental brain research*, 3(3):248, 1967.

[82] U. Köthe. Edge and junction detection with an improved structure tensor. In *Pattern Recognition*, pages 25–32. Springer, 2003.

[83] U. Köthe. Gradient-based segmentation requires doubling of the sampling rate. 2003.

[84] B. Krolla, M. Diebold, B. Goldlücke, and D. Stricker. Spherical light fields. In *BMVC*, 2014.

[85] M. F. Land and D.-E. Nilsson. *Animal Eyes.* Oxford University Press, 2012.

[86] M. Levoy and P. Hanrahan. Light field rendering. In *Proceedings of the International conference on computer graphics and interactive techniques (SIGGRAPH)*, volume 96, pages 31–42. ACM, 1996.

[87] P. Lichtsteiner, C. Posch, and T. Delbruck. A 128 $\times$ 128 120 db 15 μ s latency asynchronous temporal contrast vision sensor. *IEEE journal of solid-state circuits*, 43(2):566–576, 2008.

[88] C. Liu, S. G. Narasimhan, and A. W. Dubrawski. Matting and depth recovery of thin structures using a focal stack. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6970–6978, 2017.

[89] J. Lüke, F. Rosa, J. Marichal-Hernández, J. Sanluı, C. Domı, J. Rodrı, et al. Depth from light fields analyzing 4d local structure. *Journal of Display Technology*, 11(11):900–907, 2015.

[90] A. Lumsdaine and T. Georgiev. Full resolution lightfield rendering. *Indiana University and Adobe Systems, Tech. Rep*, 2008.

[91] A. Lumsdaine and T. Georgiev. The focused plenoptic camera. In *International Conference on Computational Photography (ICCP)*, pages 1–8. IEEE, 2009.

[92] Q.-T. Luong and O. D. Faugeras. Self-calibration of a moving camera from point correspondences and fundamental matrices. *International Journal of computer vision*, 22(3):261–289, 1997.

[93] H. Lutkepohl. Handbook of matrices. *Computational Statistics and Data Analysis*, 2(25):243, 1997.

[94] J. Marshall and J. Oberwinkler. Ultraviolet vision: The colourful world of the mantis shrimp. *Nature*, 401(6756):873, 1999.

[95] S. G. Marto, N. B. Monteiro, J. P. Barreto, and J. A. Gaspar. Structure from plenoptic imaging. In *2017 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 338–343, Sep. 2017.

[96] S. G. Marto, N. B. Monteiro, and J. A. Gaspar. Locally affine light fields as direct measurements of depth. In *RECPAD - Portuguese Conference on Pattern Recognition*, 2018.

[97] C. A. Mead and M. A. Mahowald. A silicon model of early visual processing. *Neural networks*, 1(1):91–97, 1988.

[98] V. B. Meyer-Rochow. Compound eyes of insects and crustaceans: Some examples that show there is still a lot of work left to be done. *Insect science*, 22(3):461–481, 2015.

[99] E. Miandji, S. Hajisharif, and J. Unger. A unified framework for compression and compressed sensing of light fields and light field videos. *ACM Transactions on Graphics (TOG)*, 38(3):1–18, 2019.

[100] T. Michels, A. Petersen, and R. Koch. Creating realistic ground truth data for the evaluation of calibration methods for plenoptic and conventional cameras. In *2019 International Conference on 3D Vision (3DV)*, pages 434–442. IEEE, 2019.

[101] T. Michels, A. Petersen, L. Palmieri, and R. Koch. Simulation of plenoptic cameras. In *2018-3DTV-Conference: The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON)*, pages 1–4. IEEE, 2018.

[102] N. B. Monteiro, J. P. Barreto, and J. Gaspar. Dense lightfield disparity estimation using total variation regularization. In *International Conference Image Analysis and Recognition*, pages 462–469. Springer, 2016.

[103] N. B. Monteiro, J. P. Barreto, and J. Gaspar. Surface cameras from shearing for disparity estimation on a lightfield. In *RECPAD - Portuguese Conference on Pattern Recognition*, 2018.

[104] N. B. Monteiro, J. P. Barreto, and J. A. Gaspar. Standard plenoptic cameras mapping to camera arrays and calibration based on dlt. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(11):4090–4099, 2019.

[105] N. B. Monteiro and J. A. Gaspar. Generalized camera array model for standard plenoptic cameras. In *Iberian Robotics conference*, pages 3–14. Springer, 2019.

[106] N. B. Monteiro and J. A. Gaspar. Standard plenoptic camera calibration for a range of zoom and focus levels. In *Iberian Conference on Pattern Recognition and Image Analysis*, pages 309–321. Springer, 2019.

[107] N. B. Monteiro, S. Marto, J. P. Barreto, and J. Gaspar. Depth range accuracy for plenoptic cameras. *Computer Vision and Image Understanding*, 168:104–117, 2018.

[108] N. B. Monteiro, L. Palmieri, T. Michels, L. Cruz, R. Koch, N. Gonçalves, and J. Gaspar. Geometric calibration of multi-focus plenoptic cameras. *ICCV 2019 Submission ID 2256*, 2019.

[109] P. Moon and D. E. Spencer. The photic field. *Cambridge, MA, MIT Press, 1981. 265 p.*, 1981.

[110] S. Moreschini, R. Bregovic, and A. Gotchev. Shearlet-based light field reconstruction of scenes with non-lambertian properties. In *2019 8th European Workshop on Visual Information Processing (EUVIP)*, pages 140–145. IEEE, 2019.

[111] M. K. Ng and A. C. Yau. Super-resolution image restoration from blurred low-resolution images. *Journal of Mathematical Imaging and Vision*, 23(3):367–378, 2005.

[112] R. Ng. *Digital light field photography*. PhD thesis, Stanford University, 2006.

[113] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan. Light field photography with a hand-held plenoptic camera. *Computer Science Technical Report CSTR*, 2(11):1–11, 2005.

[114] J. Nocedal and S. Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

[115] S. Nousias, F. Chadebecq, J. Pichat, P. Keane, S. Ourselin, and C. Bergeles. Corner-based geometric calibration of multi-focus plenoptic cameras. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 957–965, 2017.

[116] S. Ogata, J. Ishida, and T. Sasano. Optical sensor array in an artificial compound eye. *Optical Engineering*, 33(11):3649–3655, 1994.

[117] L. Palmieri, R. Koch, and R. O. H. Veld. The plenoptic 2.0 toolbox: Benchmarking of depth estimation methods for mla-based focused plenoptic cameras. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 649–653. IEEE, 2018.

[118] S. Parker. *Colour and Vision: Through the Eyes of Nature*. Natural History Museum, 2016.

[119] S. Pertuz, D. Puig, and M. A. Garcia. Analysis of focus measure operators for shape-from-focus. *Pattern Recognition*, 46(5):1415–1432, 2013.

[120] C. Perwass and L. Wietzke. Single lens 3d-camera with extended depth-of-field. In *Proceedings of SPIE, Human Vision and Electronic Imaging XVII*, volume 8291, page 829108. International Society for Optics and Photonics, 2012.

[121] G. Poggio. Processing of stereoscopic information in primate visual cortex. *Dynamical aspects of neocortical function*, pages 613–635, 1984.

[122] D. F. B. Portela, N. B. Monteiro, and J. A. Gaspar. Camera adaptation for deep depth from light fields. In *RECPAD - Portuguese Conference on Pattern Recognition*, 2018.

[123] Raytrix Gmbh. Raytrix RxLive Software. https://raytrix.de/downloads/. Accessed on 07.03.2019.

[124] S. Rossel. Binocular spatial localization in the praying mantis. *Journal of experimental biology*, 120(1):265–281, 1986.

[125] E. Rosten, R. Porter, and T. Drummond. Faster and better: A machine learning approach to corner detection. *IEEE transactions on pattern analysis and machine intelligence*, 32(1):105–119, 2008.

[126] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.

[127] P. Schelkens, P. Astola, E. A. Da Silva, C. Pagliari, C. Perra, I. Tabus, and O. Watanabe. Jpeg pleno light field coding technologies. In *Applications of Digital Image Processing XLII*, volume 11137, page 111371G. International Society for Optics and Photonics, 2019.

[128] G. M. Schuster, D. G. Dansereau, G. Wetzstein, and J. E. Ford. Panoramic single-aperture multi-sensor light field camera. *Optics express*, 27(26):37257–37273, 2019.

[129] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4748–4757, 2018.

[130] C. Shorten and T. M. Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):60, 2019.

[131] M. Simões, J. Bioucas-Dias, L. B. Almeida, and J. Chanussot. A convex formulation for hyperspectral image superresolution via subspace-based regularization. *Geoscience and Remote Sensing, IEEE Transactions on*, 53(6):3373–3388, 2015.

[132] K. H. Strobl and M. Lingenauber. Stepwise calibration of focused plenoptic cameras. *Computer Vision and Image Understanding*, 145:140–147, 2016.

[133] L. D. Stroebel. *Photographic materials and processes*. Focal Pr, 1986.

[134] M. W. Tao, S. Hadap, J. Malik, and R. Ramamoorthi. Depth from combining defocus and correspondence using light-field cameras. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 673–680, 2013.

[135] M. W. Tao, R. Ramamoorthi, J. Malik, and A. A. Efros. *Unified Multi-Cue Depth Estimation from Light-Field Images: Correspondence, Defocus, Shading, and Specularity*. PhD thesis, University of California, Berkeley, 2015.

[136] M. W. Tao, P. P. Srinivasan, S. Hadap, S. Rusinkiewicz, J. Malik, and R. Ramamoorthi. Shape estimation from shading, defocus, and correspondence using light-field angular coherence. *IEEE transactions on pattern analysis and machine intelligence*, 39(3):546–560, 2017.

[137] S. Vagharshakyan, R. Bregovic, and A. Gotchev. Light field reconstruction using shearlet transform. *IEEE transactions on pattern analysis and machine intelligence*, 40(1):133–147, 2017.

[138] R. Völkel, M. Eisner, and K. Weible. Miniaturized imaging systems. *Microelectronic Engineering*, 67:461–472, 2003.

[139] S. Wanner and B. Goldluecke. Variational light field analysis for disparity estimation and super-resolution. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(3):606–619, 2014.

[140] S. Wanner, S. Meister, and B. Goldluecke. Datasets and benchmarks for densely sampled 4d light fields. In *VMV*, pages 225–226. Citeseer, 2013.

[141] S. Wanner, C. Straehle, and B. Goldluecke. Globally consistent multi-label assignment on the ray space of 4d light fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1011–1018, 2013.

[142] J. Weickert. Coherence-enhancing diffusion of colour images. *Image and Vision Computing*, 17(3):201–212, 1999.

[143] G. Wetzstein, I. Ihrke, D. Lanman, and W. Heidrich. Computational plenoptic imaging. In *Computer Graphics Forum*, volume 30, pages 2397–2426. Wiley Online Library, 2011.

[144] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy. High performance imaging using large camera arrays. In *ACM Transactions on Graphics (TOG)*, volume 24, pages 765–776. ACM, 2005.

[145] B. S. Wilburn, M. Smulski, H.-H. K. Lee, and M. A. Horowitz. Light field video camera. In *Media Processors 2002*, volume 4674, pages 29–37. International Society for Optics and Photonics, 2001.

[146] M. Yamaguchi. Light-field and holographic three-dimensional displays. *JOSA A*, 33(12):2348–2364, 2016.

[147] J. Yu, L. McMillan, and S. Gortler. Scam light field rendering. In *Computer Graphics and Applications, 2002. Proceedings. 10th Pacific Conference on*, pages 137–144. IEEE, 2002.

[148] N. Zeller, F. Quint, and U. Stilla. Calibration and accuracy analysis of a focused plenoptic camera. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2(3):205, 2014.

[149] N. Zeller, F. Quint, and U. Stilla. From the calibration of a light-field camera to direct plenoptic odometry. *IEEE Journal of Selected Topics in Signal Processing*, 11(7):1004–1019, 2017.

[150] Q. Zhang, C. Zhang, J. Ling, Q. Wang, and J. Yu. A generic multi-projection-center model and calibration method for light field cameras. *IEEE transactions on pattern analysis and machine intelligence*, 2018.

[151] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22, 2000.

[152] T. Zhou, R. Tucker, J. Flynn, G. Fyffe, and N. Snavely. Stereo magnification: Learning view synthesis using multiplane images. *Proceedings of the International conference on computer graphics and interactive techniques (SIGGRAPH)*, 2018.