Leveraging blur information in plenoptic cameras : application to calibration and metric depth estimation

Mathieu Labussière

ÉCOLE DOCTORALE
DES SCIENCES POUR L'INGÉNIEUR
Université Clermont Auvergne

Leveraging blur information in plenoptic cameras:

Application to calibration and metric depth estimation

Thèse présentée par **Mathieu LABUSSIÈRE**

Pour obtenir le grade de **Docteur d'Université**

Spécialité : **Électronique et Systèmes**

Soutenue publiquement le **13 décembre 2021** devant le jury composé de

Pascal Vasseur	Président	Professeur des Universités Université de Picardie Jules Verne
Christophe Cudel	Rapporteur	Professeur des Universités Université de Haute-Alsace
Cédric Demonceaux	Rapporteur	Professeur des Universités Université de Bourgogne Franche-Comté
Peter Sturm	Examinateur	Directeur de Recherche INRIA Grenoble Rhône-Alpes
Pauline Trouvé-Peloux	Examinatrice	Ingénieure de Recherche ONERA
Omar Ait-Aider	Directeur	Maître de Conférences, HDR Université Clermont Auvergne
Frédéric Bernardin	Co-directeur	Chargé de Recherche, HDR Cerema
Céline Teulière	Encadrante	Maîtresse de Conférences Université Clermont Auvergne

Remerciements

Ce manuscrit de thèse conclu les travaux menés pendant ces trois années à l'Institut Pascal, au sein de l'équipe de vision par ordinateur ComSee. Ce voyage dans le monde de la plénoptique, qui n'a pas toujours été de tout repos, marque une étape importante dans ma vie.

Je tiens premièrement à remercier mes rapporteurs et mes examinateur·rice·s d'avoir accepté d'évaluer mes travaux, et d'avoir fait le déplacement jusqu'à Clermont-Ferrand pour ma soutenance malgré le contexte sanitaire. Pouvoir défendre en présentiel, après ces dernières années quelque peu perturbées, était important pour moi. Merci d'avoir rendu cela possible.

Je remercie évidement mon équipe encadrante : Frédéric, Omar et Céline. Merci de votre confiance pendant ces trois années, d'avoir toujours su vous rendre disponibles pour échanger et discuter pendant de longues heures, et ce, même les vendredis soirs. Je mesure la chance que j'ai pu avoir en tant que doctorant de vous avoir comme encadrants, et vous remercie encore infiniment. Je voudrais également adresser un remerciement un peu spécial à Romuald, qui a toujours été présent et m'a toujours soutenu pendant toutes mes études, et sans qui je ne serais probablement pas là aujourd'hui.

Merci à tous mes collègues et ami·e·s pour leur aide matérielle et immatérielle, que cela soit pour les manipulations, les nombreuses discussions ou bien encore tous les moments un peu moins sérieux. Sans vous ces trois années n'auraient pas eu la même saveur. Je remercie particulièrement Johann et Simon qui se reconnaîtront.

Enfin, je souhaiterais remercier les membres de ma famille, qui, même s'ils n'ont pas toujours compris ce que je faisais, ont toujours été présents pour moi.

Abstract

This thesis investigates the use of a vision sensor called a *plenoptic camera* for computer vision in robotics applications. To achieve this goal we place ourselves upstream of applications, and focus on its modelization to enable robust depth estimation.

Plenoptic or *light-field* cameras are *passive* imaging systems able to capture both *spatial* and *angular* information about a scene in a single exposure. These systems are usually built upon a micro-lenses array (MLA) placed between a main lens and a sensor. Their design enables depth estimation from a single acquisition.

The key contributions of this work lie in answering the questions "*How can we link world space information to the image space information?*" and more importantly, "*How can we link image space information to world space information?*". We address the first problem through the prism of *calibration*, by proposing a new camera model and a methodology to retrieve the intrinsic parameters of this model. We leverage blur information where it was previously considered as a drawback by explicitly modeling the defocus blur. We address the second one as the problematic of *depth estimation*, by proposing a metric depth estimation framework working directly with raw plenoptic images. It takes into account both *correspondence* and *defocus* cues. Our model generalizes to various configurations, including the multi-focus plenoptic camera (both in Galilean and Keplerian configuration), as well as to the single-focus and unfocused plenoptic camera. Our method gives accurate and precise depth estimates (a median relative error ranging from 1.27 % to 4.75 % of the distance). It outperforms state-of-the-art methods.

Having a new complete camera model and enabling robust metric depth estimation from raw images only, opens the door to many new applications. It is a first step towards practical use of plenoptic cameras in computer vision applications.

Keywords: Plenoptic camera, multi-focus, calibration, depth estimation, relative defocus blur

Résumé

Cette thèse propose d'étudier l'utilisation d'un capteur de vision appelé *caméra plénoptique* pour de la vision par ordinateur dans des applications robotiques. Plus précisément, pour atteindre cet objectif, nous nous plaçons en amont du côté applicatif, et nous nous concentrons sur sa modélisation pour permettre une estimation de profondeur robuste.

Les caméras plénoptiques ou *à champ de lumière* sont des systèmes d'imagerie *passifs* capables de capturer les informations *spatiales* et *angulaires* d'une scène en une seule exposition. Ces systèmes sont généralement constitués d'une matrice de micro-lentilles (MLA) placée entre un objectif principal et un capteur. Leur conception permet l'estimation de la profondeur à partir d'une seule acquisition.

Les contributions clés de ce travail résident dans la réponse à la question "*Comment peut-on relier l'information de l'espace monde à celle de l'espace image?*" et surtout, "*Comment peut-on relier l'information de l'espace image à celle de l'espace monde?*". Nous abordons la première par le prisme de l'*étalonnage*, en proposant un nouveau modèle de caméra et une méthodologie pour récupérer les paramètres intrinsèques de ce modèle. Nous exploitons l'information sur le flou de défocalisation là où il était auparavant considéré comme un inconvénient, en le modélisant explicitement. Nous abordons la deuxième problématique comme celle de l'*estimation de profondeur*, en proposant une méthode métrique d'estimation de profondeur fonctionnant directement avec des images brutes plénoptiques. Elle prend en compte à la fois les indices de *correspondance* et de *défocalisation*. Notre modèle se généralise à diverses configurations, y compris la caméra plénoptique multi-focales (en configuration galiléenne et keplérienne), ainsi qu'à la caméra plénoptique monofocale et non focalisée. Avec notre méthode, nous obtenons des estimations de profondeur répétables et exactes (de l'ordre de 1.27 % à 4.75 % de la distance à l'objet). Elle surpasse les résultats de l'état-de-l'art.

Le fait de disposer d'un nouveau modèle complet de caméra et de permettre une estimation métrique robuste de la profondeur à partir d'images brutes uniquement ouvre la voie à de nombreuses nouvelles applications. Il s'agit d'un premier pas vers l'utilisation concrète de caméras plénoptiques dans les applications de vision par ordinateur.

Mots-Clés : Caméra plénoptique, multi-focalisée, étalonnage, estimation de profondeur, flou relatif de défocalisation

Contents

List of Figures

List of Tables

Glossary

aperture is a hole or an opening through which light travels. More specifically, the aperture and the focal length of an optical system determine the cone angle of a bundle of rays that come to a focus in the image plane. xiv, xvii, xxiii, 3, 14, 31, 32, 36, 37, 38, 52, 63, 65, 66, 65, 69, 77, 82, 81, 82, 123, 128, 142, 151, 157, 158, 160, 164

calibration is the process of estimating the intrinsic and extrinsic parameters of a camera. v, x, xi, xiii, xiv, xvii, 4, 5, 6, 7, 21, 26, 28, 29, 30, 31, 35, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 61, 62, 63, 69, 72, 73, 74, 75, 76, 77, 78, 79, 80, 79, 80, 81, 82, 83, 84, 89, 92, 95, 98, 99, 100, 101, 109, 110, 111, 119, 122, 124, 125, 135, 136, 139, 140, 163, 164, 167, 169

circle of confusion is an optical spot caused by a cone of light rays from a lens not coming to a perfect focus when imaging a point source. It is also known as disk of confusion, circle of indistinctness, blur circle, or blur spot. xxi, 4, 30, 37, 43, 94, 95

depth of field is the distance about the plane of focus where objects appear acceptably sharp in an image. xxi, xxiii, 3, 4, 17, 30, 38, 94, 98

field of view is the extent of the observable world that is seen at any given moment. In the case of optical instruments or sensors it is a solid angle through which a detector is sensitive to electromagnetic radiation. xxi

focal length is, for an optical system, a measure of how strongly the system converges or diverges light. xvi, xix, xxiii, 15, 17, 20, 31, 32, 33, 35, 36, 37, 38, 39, 43, 44, 46, 47, 48, 49, 50, 52, 60, 61, 65, 67, 71, 77, 83, 123, 139, 151, 157, 158, 159

magnification is the process of enlarging the apparent size, not physical size, of something. This enlargement is quantified by a calculated number also called magnification. When this number is less than one, it refers to a reduction in size, sometimes called *minification* or *de-magnification*. xxiii, 33, 36, 37, 44, 52

mapping is, for an agent, the ability to construct or improve a map of its environment. 4, 39, 142

radiance is the radiant flux emitted, reflected, transmitted or received by a given surface, per unit solid angle per unit projected area. 10, 19

simultaneous localization and mapping is the computational problem of constructing or updating a map of an unknown environment while simultaneously keeping track of an agent's location within it. xxii

Acronyms

2PP two-parallel planes 19, 20, 49

AD absolute difference 131

BAP blur aware plenoptic xiv, xxiii, 4, 7, 30, 55, 60, 61, 62, 66, 69, 72, 73, 74, 75, 76, 80, 83, 84, 92, 93, 98, 121, 125, 139, 141, 142, 164, 167
BET blur equalization technique 113
BLADE blur aware depth estimation xi, xv, 7, 99, 100, 110, 113, 117, 118, 119, 121, 136
BM3D block-matching and 3D filtering 23
BOOM binarized octal orientation maps 106
BT Birchfield-Tomasi measure 103

CNN convolutional neural network 108
CoC circle of confusion 4, 30, 37, 38, 43, 48, 60, 94, 95
CPU central processing unit 136
CSAD central sub-aperture depth map xv, xviii, 130, 131
CT Census transform 104, 105

DBSCAN density-based spatial clustering of applications with noise 70, 71
DoF depth of field xxiii, 3, 4, 17, 30, 38, 94, 95, 96, 95, 98, 101, 111

EPI epipolar plane image xv, xxiii, 7, 21, 44, 100, 104, 106, 107, 108, 140

FDL-HSIFT Fourier disparity layer Harris SIFT 106
FoV field of view 142
fps frame per second 14, 15

GAN generative adversarial network 23
GPS global positioning system 3
GPU graphics processing unit 136, 141
GSS golden search section 118

HDR high dynamic range 123, 159

ICP Iterative Closest Point 142
IMU inertial measurement unit 3

lidar light detection and ranging 3, 5, 6, 7, 100, 122, 123, 124, 130, 136, 140, 167
LiFF light field features 106
Lisad light field scale and depth 107
LPQ local phase quantization 109
LSF line-spread function 141

MBE mean bias error 121
MFPC multi-focus plenoptic camera 4
MI micro-image 42, 60, 64, 65, 66, 67, 116, 118
MIA micro-images array xiv, xvii, xxiii, 14, 13, 28, 60, 62, 63, 68, 69, 94, 164
MIC micro-image center xxiii, 44, 59, 62, 64, 69, 72
MLA micro-lenses array v, vii, xiv, xvii, xxiii, 12, 13, 14, 15, 17, 28, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 55, 58, 59, 60, 62, 64, 65, 68, 70, 76, 77, 79, 83, 93, 94, 95, 96, 98, 109, 114, 116, 123, 139, 151, 157, 158, 159
MRE mean reprojection error 89
MRF Markov random field 107

NCC normalized cross-correlation 103

PIV particle image velocimetry 25, 43, 142
PnP Perspective-n-Point 50, 73, 125
PSF point-spread function xxiii, 4, 30, 37, 38, 60, 61, 75, 77, 92, 141, 164

radar radio detection and ranging 3
RMSE root-mean-square error 89, 93

SAD sum of absolute differences 103, 104, 115
SAI sub-aperture image xxiii, 7, 21, 23, 42, 43, 44, 45, 46, 48, 100, 104, 105, 106, 107, 108, 140
SfM Structure-from-Motion 26, 39, 41
SIFT scale-invariant feature transform 106, 109
SLAM simultaneous localization and mapping 26, 39, 142
SPO spinning parallelogram operator 107
SSD sum of squared differences 103
SURF speeded up robust features 109

ToF time of flight 3

Notations

The notations used in this manuscript are reported in the following tables:

Table 1 summarized the general mathematical notations;

Table 2 summarized the notations related to poses and transformations;

Table 3 summarized the algebraic distances sign convention;

Table 4 summarized the notations related to projection;

Table 5 summarized the notations related to the main lens;

Table 6 summarized the notations related to the micro-lenses array;

Table 7 summarized the notations related to the micro-images array and sensor;

Table 8 summarized the notations related to light-field;

Table 9 summarized the notations related to internal parameters;

Table 10 summarized the notations related to blur modeling;

Table 11 summarized the notations related to depth estimation.

In general, pixel counterparts of metric values are denoted in lower-case Greek letters. Bold fonts represent vectors (usually in lower-case letters) and matrices (usually in upper-case letters). Scalars are given by light letters. The distances used are algebraic distances, i.e., distances are signed according to a given convention. In this work, we will used the convention summarized in Table 3. Note that the z-axis is pointing outside the camera, i.e., in the opposite way of light propagation, as illustrated by Figure 3.14.

Table 1: General mathematical notations

Symbol	*Definition*
a, A, α	scalar such as $a \in \mathbb{K}$, with $\mathbb{K} = \mathbb{R}$ or $\mathbb{N}$
$\boldsymbol{v}$	D-dimensional vector, $\boldsymbol{v} = \begin{bmatrix} v_1 & v_2 & \cdots & v_d \end{bmatrix}^\top \in \mathbb{K}^D$
$\mathcal{S}$	set of N elements, $\mathcal{S} = \{\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_n\}$, with $\#\mathcal{S} = n$
$\boldsymbol{S}$	matrix representation of $\mathcal{S}$, $\boldsymbol{S} = \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n \end{bmatrix} \in \mathbb{R}^{D \times N}$
$\boldsymbol{u} \cdot \boldsymbol{v} = a$	dot product of two vectors (a.k.a., inner product)
$\boldsymbol{u}^\top \boldsymbol{v} = a$	inner product of two vectors (a.k.a., dot product)
$*$	convolution operator
$\circ$	element-wise matrix multiplication, Hadamard matrix product
$\oslash$	element-wise matrix division, Hadamard matrix division
∇^2	Laplacian operator
$\mathrm{f}(\cdot)$	function, e.g., log, exp, arg min
$\Theta(\cdot)$	cost function
$\mathrm{w}(\cdot)$	weight function
$\varepsilon(\cdot)$	error function
$\|\boldsymbol{v}\|_2$	Euclidean distance, or ℓ_2-norm, $\|\boldsymbol{v}\|_2 = \sqrt{\boldsymbol{v}^\top \boldsymbol{v}}$
$\|\boldsymbol{A}\|_1$	entry-wise matrix ℓ_1-norm, $\|\boldsymbol{A}\|_1 = \sum_{i,j} \lvert a_{i,j} \rvert$

Table 2: Pose and transformation notations

Symbol	*Definition*	*Space*
${}^b\boldsymbol{T}_a = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \\ \boldsymbol{0}_{1\times 3} & 0 \end{bmatrix}$	rigid transformation to express an element of the frame $\mathcal{R}_a$ in the frame $\mathcal{R}_b$, such that $\boldsymbol{p}_b = {}^b\boldsymbol{T}_a \boldsymbol{p}_a$	$\mathrm{SE}(3)$
${}^w\boldsymbol{T}_a = \boldsymbol{T}_a$	rigid transformation to world frame $\mathcal{R}_w$, $\boldsymbol{T}_a$ is called the pose of the frame $\mathcal{R}_a$	$\mathrm{SE}(3)$
$\boldsymbol{R}$	rotation matrix	$\mathrm{SO}(3)$
$\boldsymbol{t}$	translation vector	$\mathbb{R}^3$

Table 3: Convention for algebraic distances, with a being the distance to the object, b being the distance of the image, and f being the focal length of the lens.

Object Sign	*Object Type*	*Image Sign*	*Image Type*	*Focal Sign*	*Lens Type*
$a < 0$	Virtual	$b < 0$	Virtual	$f < 0$	Diverging
$a > 0$	Real	$b > 0$	Real	$f > 0$	Converging

Table 4: Projection notations

Symbol	*Definition*
$\boldsymbol{p}_w$	point from world space in homogeneous coordinates
$\boldsymbol{p}, \boldsymbol{p}_{k,l}$	projection in image space in homogeneous coordinates
$\boldsymbol{K}$	projection matrix, such that $\boldsymbol{p} = \boldsymbol{K}\boldsymbol{T}\boldsymbol{p}_w$
$\boldsymbol{K}_{\text{pinhole}}$	pin-hole projection matrix
$\boldsymbol{K}_{\text{thin-lens}}$	thin-lens projection matrix
$\boldsymbol{K}_{\text{thick-lens}}$	thick-lens projection matrix
$\Pi_{\text{k,l}}$	direct BAP projection model through the micro-lens (k, l)
$\Pi_{\text{k,l}}^{-1}$	inverse BAP projection model from the micro-lens (k, l)
$\boldsymbol{\mathcal{P}}(i, k, l)$	blur aware plenoptic projection matrix through the micro-lens (k, l) of type (i)

Table 5: Main lens parameters notations

Symbol	*Definition*	*Unit*
$\boldsymbol{T}_c$, $\left[\boldsymbol{R}_c \middle\vert \boldsymbol{t}_c\right]$	main lens pose	SE (3)
$\boldsymbol{O}$	main lens center	$\mathbb{R}^3$
F	main lens focal length	[mm]
h	main lens focus distance	[mm]
A	main lens diameter, i.e., its aperture	[mm]
N, N^*, NA	f-number, working f-number, and numerical aperture	$\mathbb{R}$
γ	magnification of the current focus setting	-
(u_0, v_0)	main lens principal point (intersection between the optical axis and the sensor plane)	[pixel^2]
(o_x, o_y)	origin of the main lens distortion	[mm]
φ	distortion function	-
$\varphi^{(r)}$	radial distortion function	-
Q_0, Q_1, Q_2	radial distortion coefficients	-
$\varphi^{(t)}$	tangential distortion function	-
P_0, P_1	tangential distortion coefficients	-
φ^{-1}	inverse distortion function	-
$Q_{\text{-1}}, Q_{\text{-2}}, Q_{\text{-3}}$	radial inverse distortion coefficients	-
$P_{\text{-1}}, P_{\text{-2}}$	tangential inverse distortion coefficients	-

Table 6: Micro-lenses array (MLA) parameters notations

Symbol	*Definition*	*Unit*
$\boldsymbol{T}_\mu,\ \left[\boldsymbol{R}_\mu \vert \boldsymbol{t}_\mu\right]$	MLA pose	SE (3)
$\boldsymbol{T}_\mu(k,l)$	pose of the micro-lens (k,l)	SE (3)
D	distance between the main lens and the MLA	[mm]
$f,\ f^{(i)}$	focal length of each type $i \in \{1,\dots,I\}$ of micro-lens	[mm]
N_μ, N_μ^*	f-number and working f-number of the micro-lens	$\mathbb{R}$
A_μ	micro-lens diameter	[mm]
Δ_μ	micro-lens inter-distance	[mm]
$\boldsymbol{C}_{k,l}$	center of the micro-lens indexed by (k,l)	$\mathbb{R}^3$
$\boldsymbol{c}_0^{k,l}$	micro-lens (k,l) principal point	[pixel^2]
$K \times L$	MLA resolution	$\mathbb{N} \times \mathbb{N}$
I	number of micro-lens types	$\mathbb{N}$
$K_1,\ K_2$	additional intrinsic parameters that account for the MLA setting	-
$a_0^{(i)}$	focus plane a micro-lens of type (i)	[mm]
$a_+^{(i)}$	far focus plane a micro-lens of type (i)	[mm]
$a_-^{(i)}$	near focus plane a micro-lens of type (i)	[mm]
$\mathrm{DOF}^{(i)}$	depth of field of a micro-lens of type (i)	[mm]

Table 7: Micro-images array (MIA) and sensor parameters notations

Symbol	*Definition*	*Unit*
$\boldsymbol{T}_s,\ \left[\boldsymbol{R}_s \vert \boldsymbol{t}_s\right]$	sensor pose	SE (3)
d	distance between the MLA and the sensor	[mm]
$D_s = D + d$	distance between the sensor and the main lens	[mm]
(τ_x, τ_y)	micro-images array translation	[pixel^2]
ϑ_z	micro-images array rotation	[rad]
$H \times W$	image resolution	$\mathbb{N} \times \mathbb{N}$
$\boldsymbol{p}_{i,j}$	image point	[pixel^2]
ϱ	pixel micro-image radius	[pixel]
R	metric micro-image radius	[mm]
δ_i	micro-images inter-distance	[pixel]
Δ_i	metric micro-images inter-distance	[mm]
$\boldsymbol{c}_{k,l}$	center of the micro-image (k,l)	[pixel^2]
s	pixel size (assumed squared)	[mm/pixel]

Table 8: Light-field related notations

Symbol	*Definition*
$\mathcal{I}, \mathcal{I}(x, y)$	2D image
$\mathcal{I}_{(i)}(x, y)$	micro-image of type (i)
$\mathcal{L}$	radiance function
$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\theta}, \nu, \tau)$	plenoptic function
$\boldsymbol{x}$	spatial position of observation $\in \mathbb{R}^3$
$\boldsymbol{\theta}$	angular direction of observation $\in \mathbb{R}^2$
ν	wavelength of the light ray $\in \mathbb{R}$
τ	time $\in \mathbb{R}$
$\mathcal{L}(s, t, u, v)$	light-field function
$\mathcal{I}_{s^*,t^*}(u, v)$	sub-aperture image (SAI)
$\mathcal{I}_{u^*,v^*}(s, t)$	light-field sub-view
$\mathcal{I}_{v^*,t^*}(u, s)$, $\mathcal{I}_{u^*,s^*}(v, t)$	epipolar plane image (EPI)
Π_f, Π_{uv}	Focal plane, orthogonal to the z-axis, indexed by (u, v)
Π_i, Π_{st}	Image plane, orthogonal to the z-axis, indexed by (s, t)

Table 9: Internal parameters notations

Symbol	*Definition*
Ξ, Ξ'	intrinsic parameters
$\{\boldsymbol{T}_c^n\}$	extrinsic parameters
λ	ratio between micro-lens diameter and micro-image diameter
α	scaling coefficient between micro-image radius and spread parameter
Ω	set of internal parameters $\{m, q'_1, \ldots, q'_I\}$
υ	virtual depth
$\{\boldsymbol{p}_{k,l}^n\}$	set of BAP features
$\{\boldsymbol{c}_{k,l}\}$	set of MIC features
$\mathcal{C}$	a cluster of features

Table 10: Blur related notations

Symbol	*Definition*
ρ	pixel blur radius
r	metric blur radius, such that $\rho = r/s$
r^*, ρ^*	radius of the smallest diffraction-limited spot
r_0, ρ_0	radius of the circle of least confusion
$\mathrm{h}(x, y)$	point-spread function (PSF)
σ	spread-parameter of the PSF
κ	blur proportionality coefficient, such that $\sigma = \kappa\rho$
$\rho_r(i, j)$	relative blur radius between micro-images of type (i) and (j)

Table 11: Depth related notations

Symbol	*Definition*
$\delta, \boldsymbol{\delta}$	disparity between micro-images
$B, \boldsymbol{B}$	baseline between micro-lenses
ψ	slope angle of EPI
$\mathcal{D}(x, y)$, $\mathcal{D}(k, l)$	depth map
$\mathcal{N}(x, y)$, $\mathcal{N}(k, l)$	neighborhood (either pixels or micro-images)
$\omega(\mathcal{I}, \boldsymbol{\delta})$	warping of image $\mathcal{I}$ at disparity $\boldsymbol{\delta}$
$\mathcal{M}, \mathcal{M}^*$	micro-image circular mask
$\mathcal{W}(\boldsymbol{p}, \mathcal{I})$	window to be extracted around $\boldsymbol{p}$ in $\mathcal{I}$
$\mathrm{std}(\mathcal{I}, \mathcal{M})$	standard deviation of the pixels intensity $\boldsymbol{p} \in \mathcal{I} \mid \mathcal{M}(\boldsymbol{p}) \neq 0$
$\Gamma(\cdot)$	scaling error function
$\gamma_0, \gamma_1, \gamma_2$	scaling error function coefficients
$\boldsymbol{P}$	point cloud matrix representation

No substance can be comprehended without light and shade; light and shade are caused by light!

– *Leonardo Da Vinci*

Chapter 1

General introduction

1.1 Context and Motivation

This work focuses on the use of *new* camera sensors called *plenoptic cameras* for computer vision in robotics applications. Although the concept is known since more than a hundred years [1], [2], implementation of such cameras is relatively new [3]–[5]. No-more restricted to custom in-house bulky prototype, they are nowadays available on the commercial market [4], [5]. Plenoptic cameras implicitly capture rich information about a scene, i.e., *spatial* and *angular* information. It means that one image can represent several points of view of a same scene. One of the main strengths of such cameras is their ability given a single snapshot to passively reconstruct a 3D representation of a scene. Indeed, they allow to gather significantly more light over a wider depth of field, and then to capture a rich 4D light-field structure providing information about textures and geometric features. With more information, the robustness of localization algorithm improves, especially during challenging weather conditions [6]. By their specific design, plenoptic cameras have a small footprint, similar to a conventional camera. This allows them to be integrated easily for the desired application, for instance, within a microscope [7] or embedded on robotics platforms [8], [9]. It also makes it possible to remove the need for a compromise between the aperture and the depth of field (DoF), the configuration being imposed by the f-number matching principle [4], [5]. Therefore, a plenoptic camera benefits from the same advantages as a conventional camera, with the

additional abilities of capturing simultaneously the visual appearance as well as the depth information about a scene. It is done from a single exposure, and without the emission of an active signal. In context of robotics applications, challenging weather conditions (especially, dust, rain, fog, snow, murky water and insufficient light) can cause even the most sophisticated vision systems to fail. These conditions can, for instance, degrade image quality or generate occlusions, which can make most of the algorithms to fail if they were not specifically developed to deal with such issues. The robustness is usually addressed by the use of other sensors such as time of flight (ToF) cameras, structured light cameras (e.g., `Microsoft Kinect`), light detection and ranging (lidar), radio detection and ranging (radar), global positioning system (GPS), inertial measurement unit (IMU), etc. But most of these sensors are active and suffer from interference, whereas a camera, which is a passive sensor, does not suffer from inter-sensor interference. The variety of applications of plenoptic cameras is great. We can for instance cite the possibility of post-processing (re-focusing), depth estimation from a single image, image noise reduction and super-resolution [10], video stabilization, isolation of obstructions [11], and specularity suppression and tolerance [12].

We believe that such capacities make the plenoptic cameras suited for applications in robotics. Under these considerations, this thesis aims at investigating the use of *plenoptic cameras* for computer vision in robotics applications (e.g., local mapping, autonomous vehicles, industrial manipulations, agricultural field, etc.). More precisely, to achieve this goal we place ourselves upstream of applications, and focus on its modelization to enable robust depth estimation. To answer the question "*How can we link world space information to the image space information?*", we will address the *calibration* problem of plenoptic cameras. As a more complex problem, the question "*How can we link image space information to world space information?*" will be addressed by the *depth estimation* problem with plenoptic cameras.

1.2 Contributions

Our contributions are twofold.

Calibration of plenoptic cameras. – *partially published as* [13] *and* [14]
We propose a new method to calibrate the multi-focus plenoptic camera within a single process taking into account all types of micro-lenses simultaneously. To exploit all available information, we propose to explicitly include the defocus blur in a new camera model. Thus, we introduce a new blur aware plenoptic (BAP) feature defined in raw image space that enables us to handle the multi-focus case. We present a new pre-calibration step using BAP features from raw

white images to provide a robust initial estimation of camera parameters. We use our BAP features in a single optimization process that retrieves intrinsic and extrinsic parameters of a multi-focus plenoptic camera directly from raw plenoptic images of a checkerboard target. In addition, we present an ablation study of the camera parameters and comparisons with state-of-the-art calibration methods. Several camera setups have been tested to validate the generalization of our method, using different focus distances and objective lenses. A simulation setup is proposed to evaluate our method on the *unfocused* configuration. Moreover, we take advantage of our BAP features to develop a new relative blur calibration process to link the geometric blur to the physical blur, i.e., the circle of confusion (CoC) to the point-spread function (PSF). This enables us to take advantage of blur in image space. Finally, we propose to use the blur to profile the plenoptic camera in terms of depth of field (DoF).

Depth estimation with plenoptic cameras. – *partially published as* [15]
We propose a new metric depth estimation algorithm using only raw images from plenoptic cameras. It is especially suited for the multi-focus configuration where several micro-lenses with different focal lengths are used. First, we introduce a metric depth estimation framework for plenoptic cameras, named blur aware depth estimation (BLADE), leveraging both spatially-variant blur and disparity cues between micro-images. It is based on area matching techniques to estimate a raw depth map $\mathcal{D}$ directly from raw plenoptic images. Two variations are considered: 1) coarse estimation, i.e., one depth per micro-image; and 2) refined estimation, i.e., one depth per pixel. Second, we include in our inverse model a depth scaling correction as we are able to measure and characterize this error. We give a methodology to correct it in a post-calibration process. Finally, we present a new dataset of 3D real-world scenes with ground truths acquired with a 3D lidar scanner, and a methodology to calibrate the extrinsic parameters. We evaluated our framework with several variations of the latter setup and against state-of-the-art methods on relative depth estimation setup.

Details on publications and communications are given in Appendix A. All our source code and datasets have been made publicly available on the GitHub of Institut Pascal, comsee-research, for reproducibility and broad accessibility, as:

libpleno is an open-source `C++` computer-vision library for plenoptic cameras modeling and processing.
Available at `https://github.com/comsee-research/libpleno`.

compote (Calibration Of Multi-focus PlenOpTic camEra) is a set of tools to pre-calibrate and calibrate (multi-focus) plenoptic cameras based on the `libpleno`.
Available at `https://github.com/comsee-research/compote`.

blade (BLur Aware Depth Estimation) is a set of tools to estimate depth map from raw images obtained by (multi-focus) plenoptic cameras based on the `libpleno`.
Available at https://github.com/comsee-research/blade.

prism (Plenoptic Raw Image Simulator) is a set of tools to generate and simulate raw images from (multi-focus) plenoptic cameras based on the `libpleno`.
Available at https://github.com/comsee-research/prism.

plenoptic-datasets is a repository containing datasets of images captured from plenoptic cameras. It includes calibration datasets `R12-A,B,C,D` obtained with a `Raytrix R12` plenoptic camera (with 50 mm and 135 mm lenses, at various focus distances), a simulated dataset `UPC-S`, for unfocused plenoptic camera (UPC) configuration, i.e., `Lytro`-like plenoptic camera, and the dataset `R12-E` containing images (obtained with a `Raytrix R12` plenoptic camera) and point clouds of 3D real-world complex scenes (obtained with a `Leica ScanStation P20`).
Available at https://github.com/comsee-research/plenoptic-datasets.

More details can be found in Appendix B and Appendix C.

1.3 Manuscript outline

In a global manner, we first give an overview of what a plenoptic camera is with some usual examples of applications. Then we present our first contribution regarding the calibration of plenoptic cameras where we explicitly model the defocus blur allowing us to provide a more complete model of the multi-focus plenoptic camera. Blur calibration is also addressed, and fills the gap between geometric blur and physical blur. Effectiveness of our method is validated by thorough experiments and comparisons with state-of-the-art methods on various configurations. Using our newly introduced model, we relate the camera parameters to the amount of blur in the micro-images, and all information can be used simultaneously, without distinction between types of micro-lenses. In a second time, we present our contribution regarding the depth estimation with plenoptic cameras. We leverage blur information where it was previously considered as a drawback, by using also defocus cues which are complementary to correspondence cues for depth estimation in our framework. We also present how to measure the depth scaling error and a methodology to correct it. Finally, we demonstrate the effectiveness of our depth scaling calibration on relative depth estimation setup and on real-world 3D complex scenes with ground truth acquired with a 3D lidar scanner. More specifically, the manuscript is organized as follows:

Chapter 2 introduces the concept of plenoptic, or light-field, imaging. We first present how to model the information available from all the light surrounding in a scene based on the *plenoptic function*. Then, we will see how to capture this information, introducing then the principle of plenoptic cameras along with a taxonomy of the different approaches. Third, we will look at how to represent and simplify the data corresponding to the light-field, reducing the plenoptic function to a four-dimensional function, mainly the two-parallel planes parametrization. Finally, we will present usual applications based on plenoptic imaging, including a focus on robotics applications.

Chapter 3 covers the problem of calibrating plenoptic cameras. We first present the theoretical foundations of modeling optics elements, their properties, and how they relate to the calibration problem. In a second time, we review the existing solutions for plenoptic camera calibration, including multi-cameras system, the *unfocused* plenoptic camera, and the *focused* plenoptic camera. We will see that current calibration methods rely on simplified projection models, use features from reconstructed images, or require separated calibrations for each type of micro-lens, which is not satisfactory especially when dealing with the multi-focus plenoptic camera. We will then present our solution for the calibration of plenoptic cameras, by introducing a new projection model and its inverse, leveraging our newly introduced blur aware plenoptic (BAP) features. We also include results regarding the calibration of the relative blur in our method. Finally, thorough evaluations of our method are presented and discussed.

Chapter 4 covers the problem of depth estimation from single images acquired with plenoptic cameras. First, we review the existing methods for depth estimation based on light-field data. We will see that most of them are working with sub-aperture images (SAIs) or epipolar plane images (EPIs) which is prone to error as depth is usually required to reconstruct the light-field or SAIs – in the focused plenoptic camera case. To overcome this issue, algorithms can work directly with raw plenoptic images, at micro-images level. However, usually only micro-images with the smallest amount of blur are used, or alternatively, specific patterns are designed to exploit the information. In opposition, we will see that using our camera model, we relate the camera parameters to the amount of blur in the image, and all information can be used simultaneously, without distinction between types of micro-lenses. We propose then to leverage blur information where it was previously considered as a drawback. Second, we explain how we link the disparity in image space to the defocus blur information. Indeed defocus cues are complementary to correspondence cues, and can improve the quality of depth estimation. Third, we detail our blur aware depth estimation (BLADE) framework and the depth calibration process. Indeed, we will see that the inverse projection has a depth scaling error. We

present then a methodology to measure this error and to correct it. Finally, our experimental setups are presented and our results are given and discussed on relative depth estimation setup and on real-world 3D complex scenes with ground truth acquired with a 3D lidar scanner.

Chapter 5 provides a general conclusion with discussions about our contributions and the perspectives for improvements and future works leveraging both our new model and our framework for depth estimation and 3D reconstruction.

Chapter 2

Overview of plenoptic imaging

Introduction

This chapter introduces the concept of plenoptic, or light-field, imaging. We first present how to model the information available from all the surrounding light in a scene. Then, we show how to capture this information, introducing the principle of plenoptic camera. Third, we show how to represent and simplify the data corresponding to the light-field. Finally, we present usual applications based on plenoptic imaging.

2.1 The plenoptic function

The word *plenoptic* is built upon the Latin root *plenus*, meaning *whole, complete*, and from the word *optic*. Plenoptic imaging, also called *light-field* imaging, is a technology aiming at capturing a maximum of visual information from the surrounding environment. Primary works related to this technology emerged from the beginning of the XX$^{\text{th}}$ century, and was first introduced under the terminology *Integral Photography* by Lippmann [2], [16]. It takes inspiration from the *parallax stereogram* of Ives [1]. The term *light-field* was used for the first time only thirty years later by the mathematician Gershun [17] in his work about light properties in three-dimensional space. However, this work remained purely theoretical until the introduction of the *plenoptic function* by Adelson and Bergen [18].

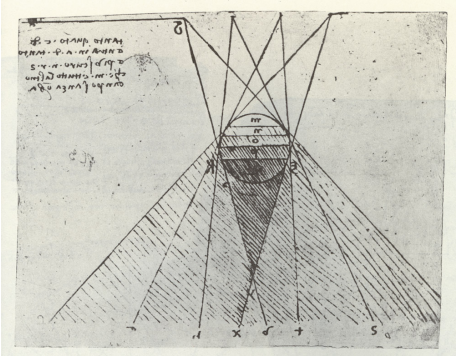


Figure 2.1: Diagram of a sphere illuminated by light falling through a window as sketched by Leonardo da Vinci, illustrating the concept of *pyramid of light*.

A *light ray* is an electromagnetic wave propagating in a straight line in homogeneous media, in a specific direction over time, and characterized by its wavelength. As early as the XVI$^{\text{th}}$ century, Leonardo da Vinci was interested in the distribution of light rays in space from the objects to the eye, which he named *pyramid of light* (as illustrated in Figure 2.1). It was not until the end of the XX$^{\text{th}}$ century that Adelson and Bergen [18] proposed a new function to model the temporal evolution of the set of all light rays emanating from all points in space, in all directions at all wavelengths. Mathematically, this function is a seven-dimensional function defined as

$$\mathcal{L}\left(\boldsymbol{x}, \boldsymbol{\theta}, \nu, \tau\right), \tag{2.1}$$

where $\boldsymbol{x} \in \mathbb{R}^3$ is the *spatial* position of observation, $\boldsymbol{\theta} \in \mathbb{R}^2$ is the *angular* direction of observation, ν is the wavelength of the light ray and τ is the time. This function expresses the radiance of each light ray, and forms the so-called *light-field*.

The purpose of an imaging system is to map these incoming light rays from a scene onto pixels of photo-sensitive detectors. Each pixel collects radiance from a bundle of closely packed rays in a non-zero aperture size system. This bundle can be represented by a single chief (or principle) ray when studying the geometric properties of the imaging system. Imaging systems allow to capture only a part of the plenoptic function, as summarized in Table 2.1. In case of a conventional camera, the sensor is only able to capture rays emanating from one point of view at a given instant. If we consider gray-level image, the function is partially evaluated as $\mathcal{L}\left(\boldsymbol{x}\right)$. If we consider now color image, several wavelengths can be captured, and the

Table 2.1: Imaging systems with dimensions of the plenoptic function that can be retrieved.

Imaging system	**Spatial** ($\boldsymbol{x}$)	**Angular** ($\boldsymbol{\theta}$)	**Temporal** (τ)
conventional camera	✓	-	-
video camera	✓	-	✓
plenoptic camera	✓	✓	-
plenoptic video camera	✓	✓	✓

function is expressed as $\mathcal{L}(\boldsymbol{x}, \nu)$. The frequency corresponds then to the available color channels: a grayscale camera only captures one discrete value of this function; a color camera captures three discrete channels and therefore more information from the function; finally, multi-spectral, x-rays, thermal, etc. cameras capture different values of the function for the given frequencies (note that for an RGB image, we can model it as three distinct functions associated to each discrete wavelength, and thus return to the previous case). If we consider several color images (e.g., in video mode) and then add the *temporal* dimension, the function is now $\mathcal{L}(\boldsymbol{x}, \nu, \tau)$. However, it is not possible to capture several points of view from a single acquisition. The *angular* information $\boldsymbol{\theta}$ cannot be retrieved, and part of the light-field is lost.

2.2 Plenoptic imaging systems

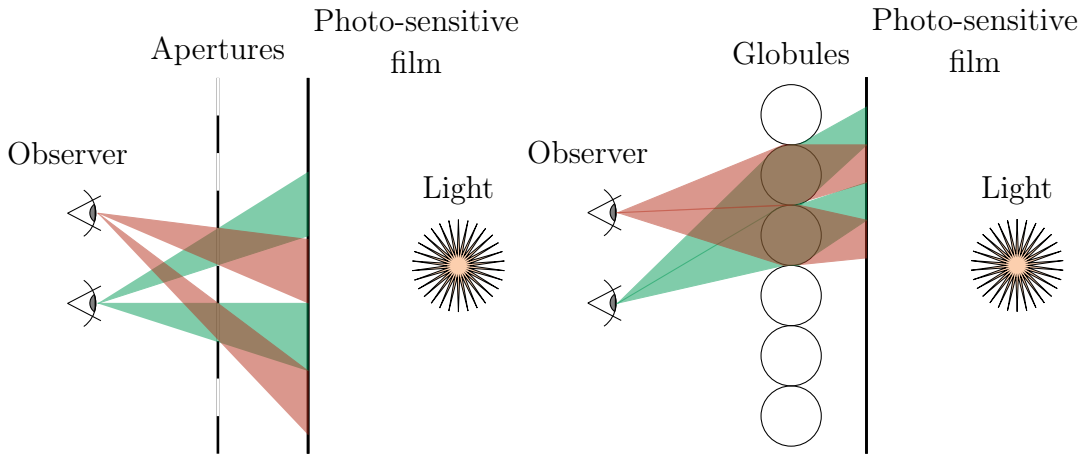
A *plenoptic* camera, or *light-field* camera, is an imaging system that allows to retrieve *spatial* as well as *angular* information from the plenoptic function, i.e., $\mathcal{L}(\boldsymbol{x}, \boldsymbol{\theta})$. The resolution of plenoptic imaging is expressed as `Spatial` × `Angular`, where `Spatial` is the resolution `width` × `height`, and `Angular` is the number of points of view. From *Lumigraph* [16] in the early XX$^{\text{th}}$ century to commercial *plenoptic cameras* [4], [5] nowadays, several designs have been proposed and are available to the public.

2.2.1 Timeline

2.2.1.1 First approaches and concepts

In 1903, Ives [1] patented a new device (illustrated in Figure 2.2) composed of a parallax barrier enabling the user to receive one different image on each eye, thus creating the impression of relief. In 1908, Lippmann [2], [16] presented the first concrete plenoptic camera, which he named integral photography. His system is based on a matrix of glass spheres, named *globules*, placed in front of a photo-

sensitive film (illustrated in Figure 2.2 and Figure 2.3). In 1930, Ives [19] proposed an improved version named the *parallax panoramagram*, based on a main lens acting as an objective, and an array of holes placed in between the lens and the film.



(a) The *parallax stereogram* of Ives [1] (b) The *integral photography* device of Lippmann [2]

Figure 2.2: First approaches to capture the light-field.

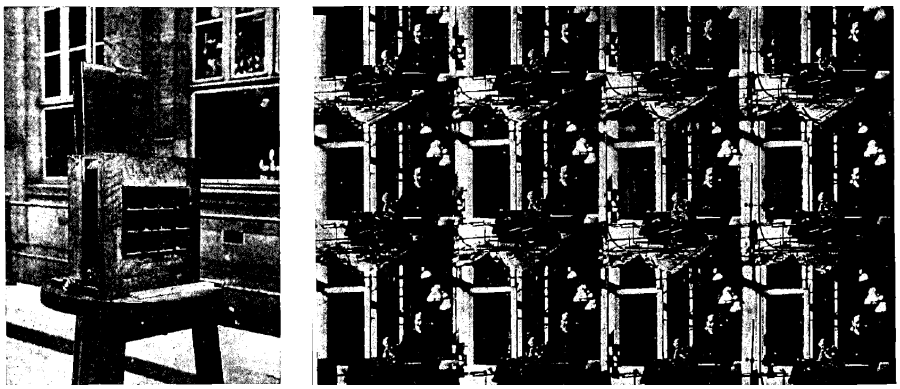


Figure 2.3: *Left:* the camera built by Lippmann in 1911 based on his concept of integral photography; *Right:* an example of resulting image. [16].

2.2.1.2 Towards the plenoptic cameras

The first compact plenoptic camera model was presented by Adelson and Wang [3]. His design is based on a micro-lenses array (MLA) placed between a photo-sensitive film (i.e., the sensor) and a main lens. This is usually referred as *lenslet-based* plenoptic cameras in the literature. In the following, when using the terminology *plenoptic camera*, we will refer to such a design unless otherwise stated. This first compact camera is thus able to record more information from the light-field than conventional cameras. However, Adelson and Wang did not build the camera, but only prototyped a non-portable version containing a relay lens. Thereafter, several

(a) `Lytro Illum` camera [4] (b) `Raytrix R12` camera [5]

Figure 2.4: Example of commercial plenoptic cameras available on consumer and industrial grades market.

commercial plenoptic cameras (Figure 2.4) have been developed targeting either the consumer market or industrial applications.

In 2005, the former company `Lytro`[1] is one of the first commercial companies proposing a plenoptic camera with consumer market application, especially in photography. One of their models is illustrated in Figure 2.4a. Their camera is built upon the thesis work of Ng *et al.* [4], which proposes a similar design to the one of Adelson and Wang [3], but simplified, less bulky and of smaller size allowing to aim at the photography market.

In 2012, the company `Raytrix GmbH`[2] proposed several plenoptic camera models – one example is illustrated in Figure 2.4b –, based on the work of Lumsdaine and Georgiev [20]–[22], both working for the company `Adobe`. Unlike `Lytro`, which initially targeted the consumer market, the main market of `Raytrix`'s cameras is industrial and scientific applications.

2.2.2 Taxonomy

The previous devices are part of the *multiplexing imaging systems*, as they mapped the *spatial* and *angular* information, i.e., a four-dimensional information, into a two-dimensional image. Both types of information are multiplexed onto the sensor

[1] The company `Lytro` was founded in 2006 by Ren Ng. Initially proposing optical systems to carry out plenoptic photography (`Lytro Illium`), the company turns to the market of the virtual reality (`Lytro Immerge`). Interested by this technology, `Google` makes the acquisition of this company which thus ceases its activities in March 2018.

[2] The company `Raytrix GmbH` is a German company founded in 2010 by Christian Perwass and Lennart Wietzke, that created and marketed the first commercial plenoptic cameras with high resolution. See https://raytrix.de/ (accessed on the 27$^{\text{th}}$ of August 2021).

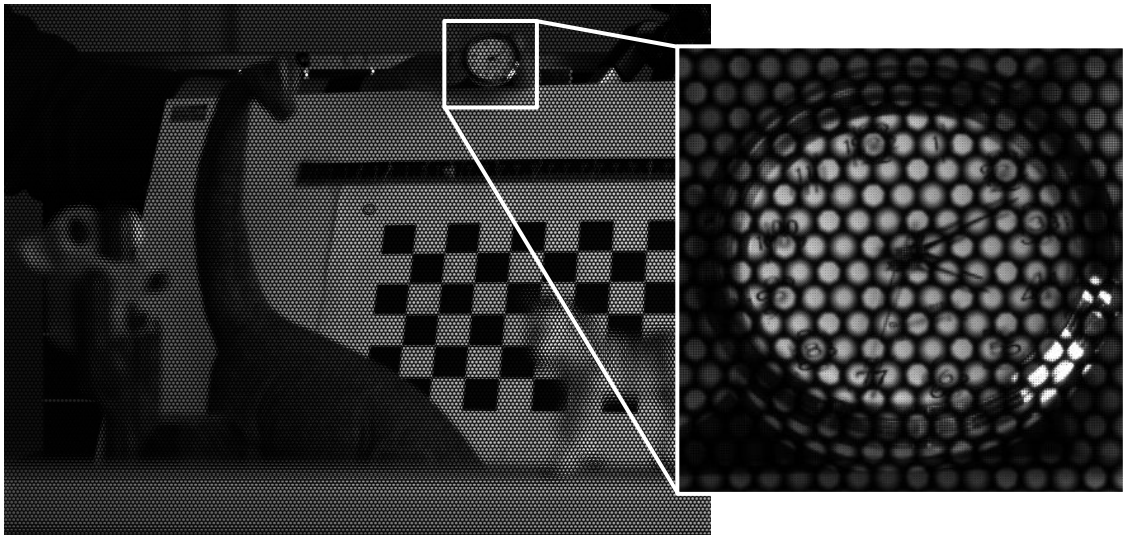


Figure 2.5: Example of raw plenoptic image multiplexing both *angular* and *spatial* information onto the sensor in the form of a micro-images array (MIA) with several types of micro-lenses, thus different amounts of blur. Image taken with a `Raytrix R12` camera, as illustrated by Figure 2.4b.

in the form of a micro-images array (MIA), as shown in Figure 2.5. However, this implies a trade-off between the angular and spatial resolutions [22]–[24]. It is balanced according to the MLA position with respect to the focal plane of the main lens and the sensor plane, corresponding to *unfocused* [4] or *focused* [5], [25] configurations (see Figure 2.7).

Other approaches not based on MLA exist to acquire the light-field. Usually, we can consider two other categories of approaches to capture the light-field:

Multi-sensors imaging. It corresponds to a setup of sensors such that we can capture several points of view from a same scene at the same time. *Spatial* resolution is given by the sensor resolution, and *angular* resolution by the number and the arrangement of these sensors. For instance, Stanford University developed a large array of cameras allowing high performance imaging with a resolution of $640 \times 480 \times 10 \times 10$ at 30 frame per second (fps) [26]. However, this kind of system is intrinsically large, bulky and costly regarding the number of sensors required. A compact small-sized model, i.e., of the size of a coin, of this technology has been proposed by the company `Pelican Imaging` in order to be integrated in smartphones [27]. For example, the `PiCam` has a resolution of $1000 \times 750 \times 4 \times 4$.

Time-sequential imaging. It corresponds to the use of a unique sensor but under different exposure setups which captures several images at several instants. The combination of those images allows to reconstruct the light-field. For instance, it is possible to use a gantry system to control the position of a

camera to acquire several scenes [28]. We can also use technologies based on programmable aperture, e.g., coded-aperture cameras, which allow to capture light rays coming from a specific direction, but require several acquisitions over time to vary the aperture shape [29]. Although cheaper as it required only one sensor, this kind of approach needs to know the camera position very precisely at each instant to associate the data. Furthermore, the process is usually time-consuming.

Based on [30], Table 2.2 summarizes the different kinds of approaches to acquire the light-field, with some examples of imaging systems. For more details, the reader can refer the complete overview of Wu *et al.* [30]. In the following, we will focus solely on *multiplexing imaging systems*, such as cameras based on a MLA placed between a main lens and a sensor.

2.2.3 Unfocused plenoptic camera

Each version of the plenoptic camera captures the light-field in a different way. The first plenoptic system, called *plenoptic* `1.0`, *unfocused plenoptic camera* or *standard plenoptic camera*, corresponds to the model of Adelson and Wang [3] later studied by Ng *et al.* [4]. It is characterized by a main lens, a sensor, and a MLA placed at a distance equal to the focal length of the micro-lenses (see Figure 2.7b). The micro-lenses are then focused at infinity, hence the *unfocused* designation, meaning that the main lens focuses light in the micro-lenses plane. In this configuration, the *spatial* resolution of a reconstructed image of the scene is given by the number of micro-lenses in the array. The *angular* resolution is given by the number of pixels behind each micro-lens. In other words, each micro-lens captures a point in space, and each pixel under this micro-lens encodes the orientation of the light ray emitted from this point in space.

Although having a poor *spatial* resolution, most of the available prototypes of plenoptic cameras are based on this design. Indeed, it is easier to reconstruct the light-field from raw images, and it thus simplifies the use in real life applications.

2.2.4 Focused plenoptic camera

In order to improve the *spatial* resolution, a trade-off has to be made with the *angular* resolution. A new design of plenoptic camera then emerged, called *plenoptic* `2.0` or *focused plenoptic camera*. It corresponds to the model of Lumsdaine and Georgiev [20]–[22]. Unlike the *unfocused* plenoptic camera, the MLA is no longer placed at a distance equal to the focal length of the micro-lenses. Two configurations can then be considered:

Table 2.2: Summary of some approaches to capture the plenoptic function.

	Reference	*Year*	*Implementation*	*Resolution*	*Speed* [fps]
Multi-sensors	Yang *et al.* [31]	2002	8×8 cameras array	$320 \times 240 \times 8 \times 8$	15-20
	Zhang and Chen [32]	2004	6×8 cameras array	$320 \times 240 \times 6 \times 8$	15-20
	Wilburn *et al.* [26]	2005	10×10 cameras array	$640 \times 480 \times 10 \times 10$	30
	`PiCam` [27]	2013	4×4 cameras array	$1000 \times 750 \times 4 \times 4$	-
	Lin *et al.* [33]	2015	5×5 cameras array	$1024 \times 768 \times 5 \times 5$	30
Sequential	`LF Lego Gantry` *	2002	gantry	$1024 \times 1024 \times 17 \times 17$	1/18 000
	Kim *et al.* [28]	2013	linear gantry	$5616 \times 3744 \times 100 \times 1$	1/120
	Liang *et al.* [29]	2008	programmable aperture	$3039 \times 2014 \times 5 \times 5$	2
Multiplexing	Ng *et al.* [4]	2005	MLA	$292 \times 292 \times 14 \times 14$	62.5
	Georgiev *et al.* [23]	2006	Lens & prisms	$700 \times 700 \times 4 \times 5$	-
	Levoy [34]	2006	MLA	$120 \times 120 \times 17 \times 17$	15
	`Raytrix` [5]	2012	MLA	>1 Mpix (effective)	15-180
	`Lytro Illum` [35]	2013	MLA	$625 \times 434 \times 15 \times 15$	3
	Riou *et al.* [36]	2015	2×2 multi-lenses	$550 \times 550 \times 2 \times 2$	-

* *The (New) Stanford Light Field Archive*, Computer Graphics Laboratory, Stanford University: http://lightfield.stanford.edu/lfs.html (last accessed the 30th of August 2021)

1. the *Galilean* configuration, where the MLA is placed in front of the focal plane of the main lens, creating a virtual image behind the sensor. The micro-lenses are focused on a virtual intermediate plane (see Figure 2.7d).

2. the *Keplerian* configuration, where the MLA is placed behind the focal plane of the main lens, creating a real image in front of the sensor. The micro-lenses are focused on a real intermediate plane (see Figure 2.7c).

These different designs are illustrated in the Figure 2.7. For comparison between unfocused and focused plenoptic cameras, the readers can refer to the work of Zhu *et al.* [37] and to the technical report of Cossu *et al.* [38].

Multi-focus plenoptic camera. To further improve the depth of field (DoF) of the *focused* plenoptic camera, Perwaß and Wietzke [5] propose to use an MLA with different types of intertwined micro-lenses, each one having its own focal length carefully chosen such that their DoFs just touched. It generates different amounts of blur as illustrated in Figure 2.6. More details will be provided in Section 3.3.

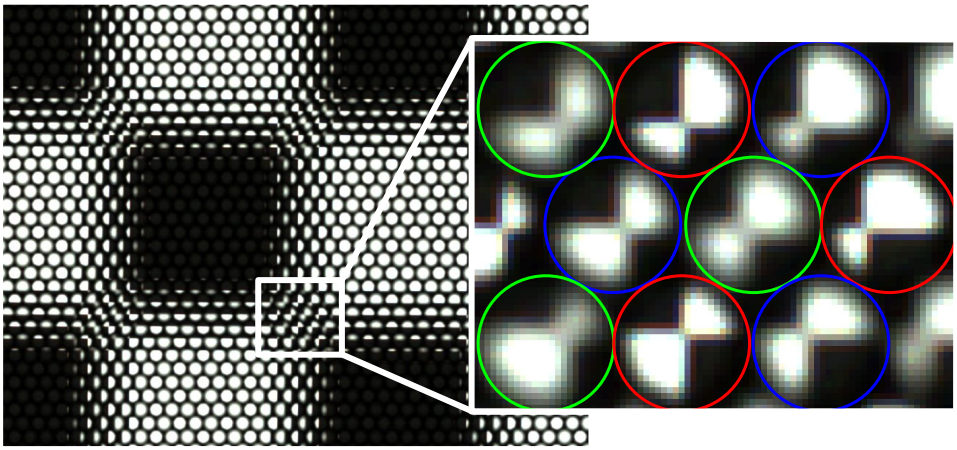
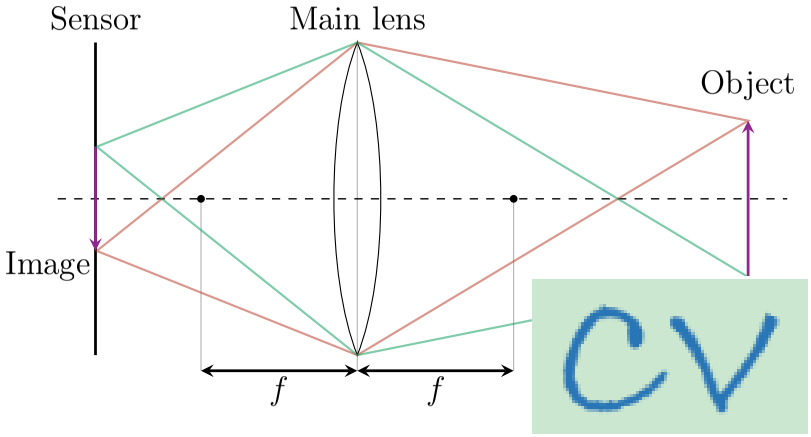


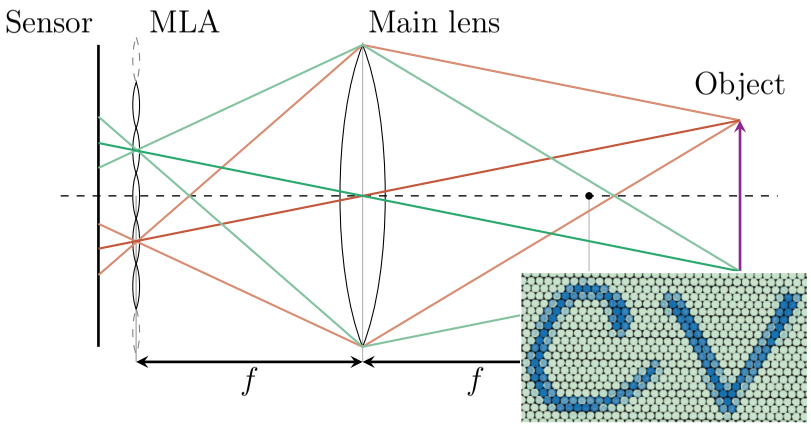
Figure 2.6: Example of raw multi-focus plenoptic image of a checkerboard with several types of micro-lenses, thus different amounts of blur. Image taken with a `Raytrix R12` camera, as illustrated in Figure 2.4b.

2.3 Light-field representation

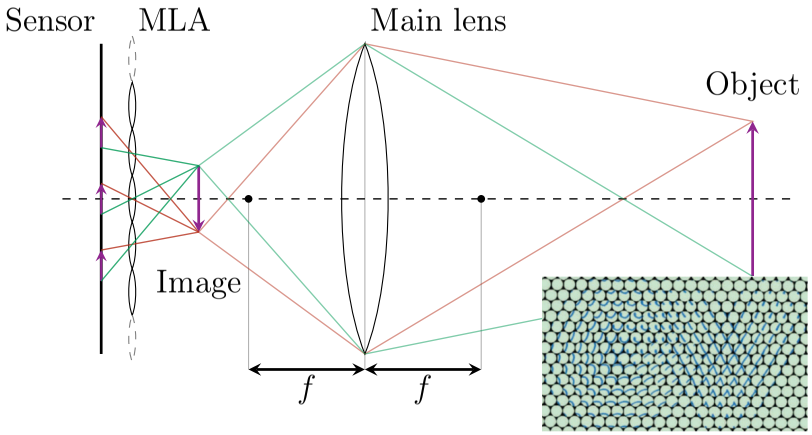
The plenoptic function is an ideal model, but in practice it is not easy to manipulate a seven-dimensional object. In particular, when capturing the plenoptic function with an imaging system, the light-field is expressed with redundancy. Simplified parametrization can be used to approximate the light-field, and to reduce the dimensionality.



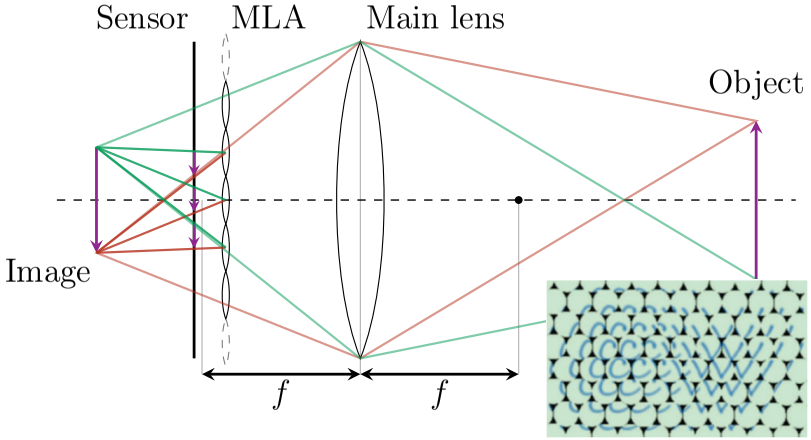
(a) Conventional camera



(b) Unfocused design (1.0)



(c) Keplerian design (2.0)



(d) Galilean design (2.0)

Figure 2.7: Comparison of optical designs of a conventional camera and plenoptic cameras.

2.3.1 Parametrization

In plenoptic imaging, the light-field is considered as a representation of the scene. McMillan and Bishop [39] proposed a simplified parametrization of the light-field under the following hypotheses:

1. the scene is supposed *static* and *lambertian* during the acquisition, i.e., light rays propagate freely in space and do not vary according to time, meaning that τ can be seen as constant;
2. lighting is supposed *invariant*, and can be approximated by monochromatic light (in practice, three discrete color channels are used), meaning that ν can be seen as constant.

Therefore, the following parametrization,

$$\mathcal{L}\left(\boldsymbol{x}, \boldsymbol{\theta}, \nu, \tau\right)\Big|_{\nu=\text{cte},\tau=\text{cte}} = \mathcal{L}\left(\boldsymbol{x}, \boldsymbol{\theta}\right), \tag{2.2}$$

is only function of the observation position $\boldsymbol{x}$ and of the light ray orientations $\boldsymbol{\theta}$. Levoy and Hanrahan [40] highlighted that the parametrization still contains redundancy under the following additional hypothesis

3. the radiance does not vary along the light ray, meaning that one spatial dimension can be reduced.

So, they proposed to model the light-field by a four-dimensional function based on the parametrization of the light rays going through two distinct parallel planes, i.e.,

$$\mathcal{L}\left(s, t, u, v\right), \tag{2.3}$$

where (s, t) are the coordinates in a first plane Π_{st}, and (u, v) are the coordinates in a second plane Π_{uv}. The two planes are separated by a certain distance, usually set to 1. Around the same time, Gortler *et al.* [41] also proposed a simplified similar model named the *Lumigraph*. It is a three-dimensional cube containing the whole scene, allowing to parametrize the light rays by their intersection with two faces of this cube. Following the same idea, the four-dimensional light-field, encoding the set of light rays, can be parametrized in several ways, each one having its pros and cons with respect to the desired application. It mostly influences the sampling of the light-field for rendering. The readers can refer to [42] for more details. Some of these parametrizations are summarized in Table 2.3 and illustrated in Figure 2.8. In addition to those, we can also cite the alternative parametrizations as the one of Isaksen *et al.* [43] allowing to dynamically re-parameterize the light-field, or the non-structured parametrization of [44]. Moreover, Alain and Smolic [45] studied the

spectral properties of re-parameterized light field, focusing on the two-parallel planes (2PP) but also providing theoretical analysis not restricted to parallel planes. In the following, unless otherwise stated, the default parametrization is the 2PP.

Table 2.3: Summary of some possible parametrization of the light-field.

Name	*Explanation*	*Ref.*
2 Parallel Planes (2PP)	(*the most used parametrization in the literature*) each ray is parameterized by its intersections with two parallel planes	[40]–[42]
Spherical (2SP)	parameterized by the intersection with two spheres, the first encoding the position, and the second encoding the direction placed at the intersection of the ray with the first sphere	[46]
Sphere-Sphere (SSP)	parameterized by two intersections on a same sphere	[47]
Sphere-Plane (SPP)	parameterized by the intersection between a plane and the normal to the plane chosen such that it is perpendicular to the ray and passes through the center	[47]
Polar coordinates	parameterized by the point on the ray closest to the center and then using the polar angles, the distance, and the rotation of the ray within the tangent plane	[48]

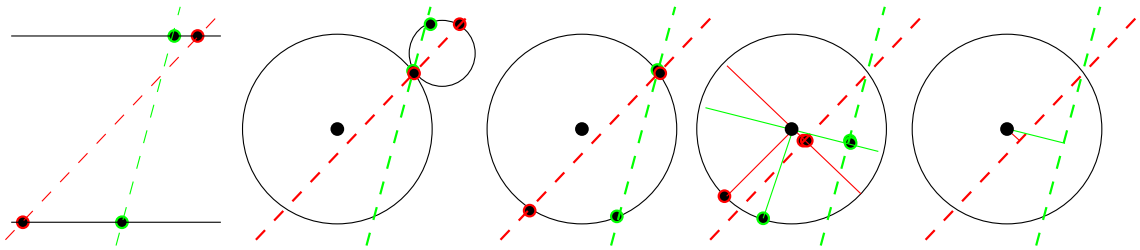


Figure 2.8: Five light-field parametrizations from the literature [42]. Two distinct rays are illustrated in dashed green and dashed red. From left to right: 2-Parallel Planes (2PP), Spherical (2SP), Sphere-Sphere (SSP), Sphere-Plane (SPP), and Polar coordinates.

2.3.2 Visualization

We have seen that the plenoptic function can be reduced to a four-dimensional representation. However, the dimensionality is still too high to be visualized easily by human eye. Several representations have been proposed to visualize the data in an intelligible way. Considering the 2PP parametrization of the light-field,

i.e., $\mathcal{L}(s,t,u,v)$, usually the plane Π_{st} can be interpreted as a set of cameras having their focal lengths on the plane Π_{uv}. Meaning that one can think of the (s,t) plane as selecting a camera, and (u,v) as selecting a pixel. From this model, we can draw the following two interpretations:

- Each camera records rays going through Π_{uv} and focusing on a single point in the plane Π_{st}, i.e., it corresponds to one specific point of view. The image obtained for fixed coordinates (s^*,t^*), $\mathcal{I}_{s^*,t^*}(u,v)$ is called sub-aperture image (SAI) or pinhole view. The light-field can thus be represented as a multi-views array, containing those SAIs, as illustrated in Figure 2.9a.
- Each point on the plane Π_{uv} represents the set of all rays going through Π_{st}, i.e., it corresponds to the same point seen from different points of view. The image obtained for fixed coordinates (u^*,v^*), $\mathcal{I}_{u^*,v^*}(s,t)$ is called light-field sub-view. The light-field can thus be represented as the set of all points of view associated to each point as illustrated in Figure 2.9b.

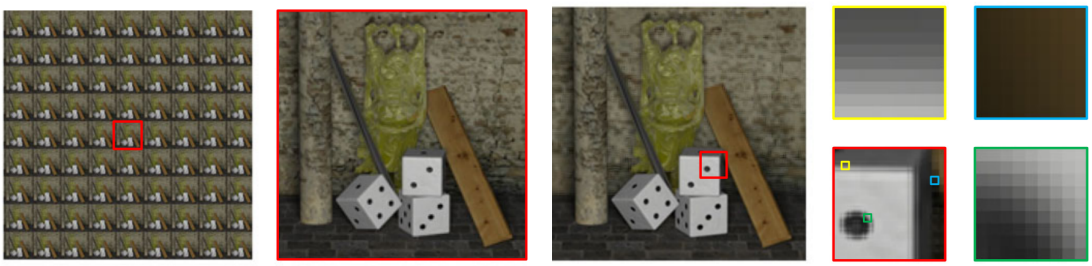
Both previous interpretations are packing together either both *spatial* or both *angular* dimensions. We can also mix spatial and angular information to produce images slices, $\mathcal{I}_{v^*,t^*}(u,s)$ or $\mathcal{I}_{u^*,s^*}(v,t)$, usually called EPIs [49], and illustrated in Figure 2.9c. Those representations capture both angular and spatial information, but also encode depth information based on variation in this space, such as the slope of the lines.

2.4 Applications

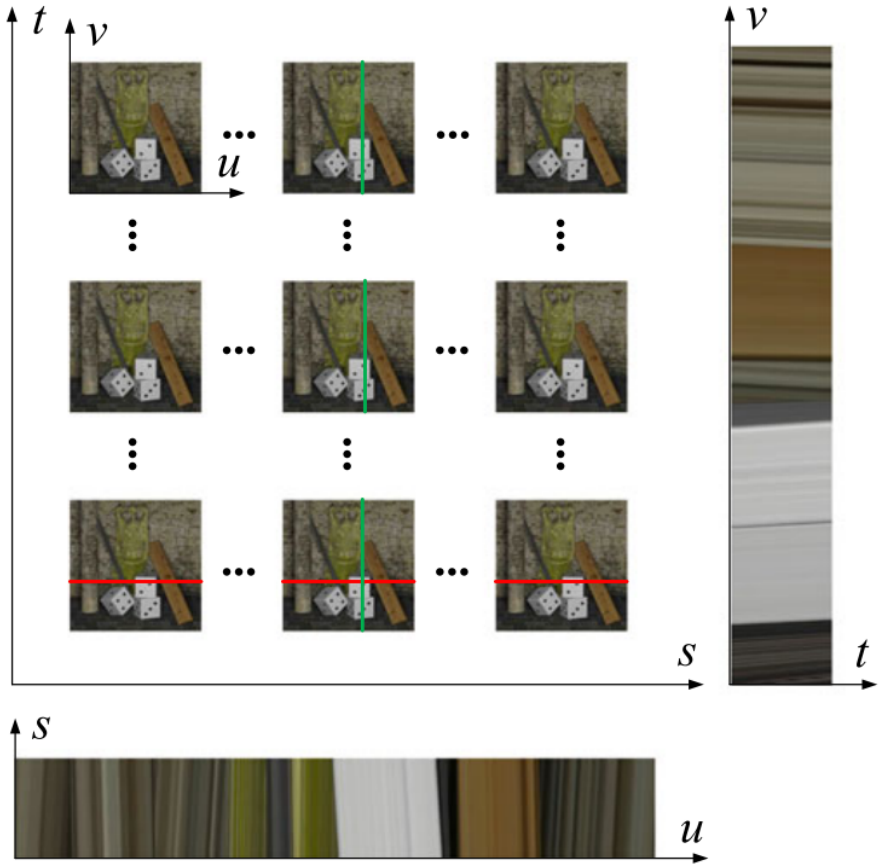
Several applications can leverage both *spatial* and *angular* information captured by a plenoptic camera. This additional information can play a significant role in improving computer vision applications. Such applications, however, required precise calibration of the camera parameters, which will be addressed in the next chapter.

2.4.1 Rendering, denoising, and super-resolution

Image-based rendering. Rendering is usually done by approximating the plenoptic function to render a novel set of two-dimensional images from other images. McMillan and Bishop [39] presented an image-based rendering system based on sampling, reconstructing, and re-sampling of the plenoptic function. Chan and Shum [50] studied the sampling and reconstruction problem of plenoptic function using spectral analysis to derive spectral support of the light-field. [51] adopted a geometric



(a) (b)



(c)

Figure 2.9: Several visualizations of the light-field. (a) Multi-views array of SAIs; in the red square, a view corresponding to fixed coordinate (s, t). (b) Light-field sub-views; for three points (yellow, blue and green) are represented the set of rays depending on the point of view. (c) EPIs views; for the red line, the horizontal EPI is given for fixed (v, t) coordinates; for the green line, the vertical EPI is given for fixed (u, s) coordinates. Images from [30].

approach to investigate the minimum sampling problem for light-field rendering, with and without geometry information of the scene. A comprehensive review up to the date of 2016 was provided by Ihrke *et al.* [52] and aimed at revisiting 25 years of research in light-field imaging. Hog *et al.* [53] presented an image rendering pipeline tailored for focused plenoptic cameras. Their algorithm does not need to generate SAIs or EPIs. Rendering as well as editing have been addressed in his thesis [54]. Comparison of reconstruction approaches between unfocused and focused plenoptic imaging systems has been done in [55]. Filipe *et al.* [56] proposed an improved patch-based rendering of all-in-focus images for focused plenoptic camera.

Light-field denoising. Dansereau *et al.* [57] described a 4D frequency-planar filter constructed using two frequency-hyperplanar filters arranged in a cascaded configuration. Alain and Smolic [58] extended the state-of-the-art block-matching and 3D filtering (BM3D) image denoising filter to light-fields. It demonstrates an improvement of the light-field quality. Allain *et al.* [59] presented a novel light-field denoising algorithm using 4D anisotropic diffusion in ray space. It does not require prior estimation of disparity maps.

Super-resolution: spatial and angular. We can apply super-resolution either in the *spatial* or in the *angular* dimensions. Light-field spatial super-resolution typically uses depth information, estimated from the data, to super-resolve a view by propagating light rays intensity values from neighboring views to sub-pixel positions, as proposed in [60]–[63]. Angular super-resolution is typically used to synthesize virtual viewpoints from a small set of views, as proposed in [64]–[67]. A comparative study is available in [10].

Synthetic aperture imaging. Light-field provides sufficient information for post-focus capability. Refocus can be understood as virtually sliding the camera focus plane to a different plane, as illustrated in Figure 2.10. Synthetic aperture imaging can blur out scene elements which fall outside the plane of focus [4], [43]. Frequency planar filter [43], 4D planar filter [68], shift and sum filter [69] are capable of focusing on object at particular depth. Volumetric filtering capable of selecting objects over range of depths is discussed in [70]. It is possible to select multiple depth planes to be in focus and to create an *all in-focus* rendered view, as in Figure 2.11.

Compressing. The amount of data captured by a plenoptic camera is substantial, especially in terms of memory space. Indeed, light-field processing requires a large memory bandwidth and more computational power and time, due to the high dimensionality of the data. Some compressing techniques have then been developed to address this issue, such as adaptations for conventional compressing techniques, e.g., `JPEG PLENO` [71], `MPEG-I` [72], `H.224` [73], `HEVC` [74], `3D-DCT` [75] or `3D-DWT` [76]. Recently, Jia *et al.* [77] proposed a new approach based on learning, in particular

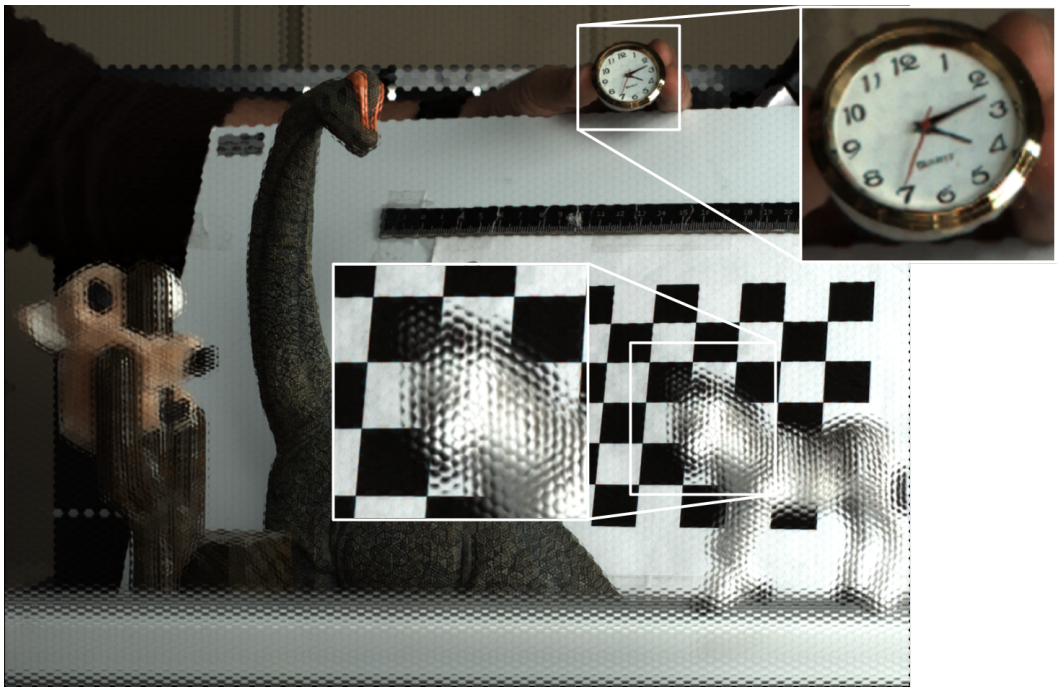


Figure 2.10: Example of post-refocusing with a plenoptic camera, at the distance corresponding to the watch plane. Corresponding raw-image is given in Figure 2.5.

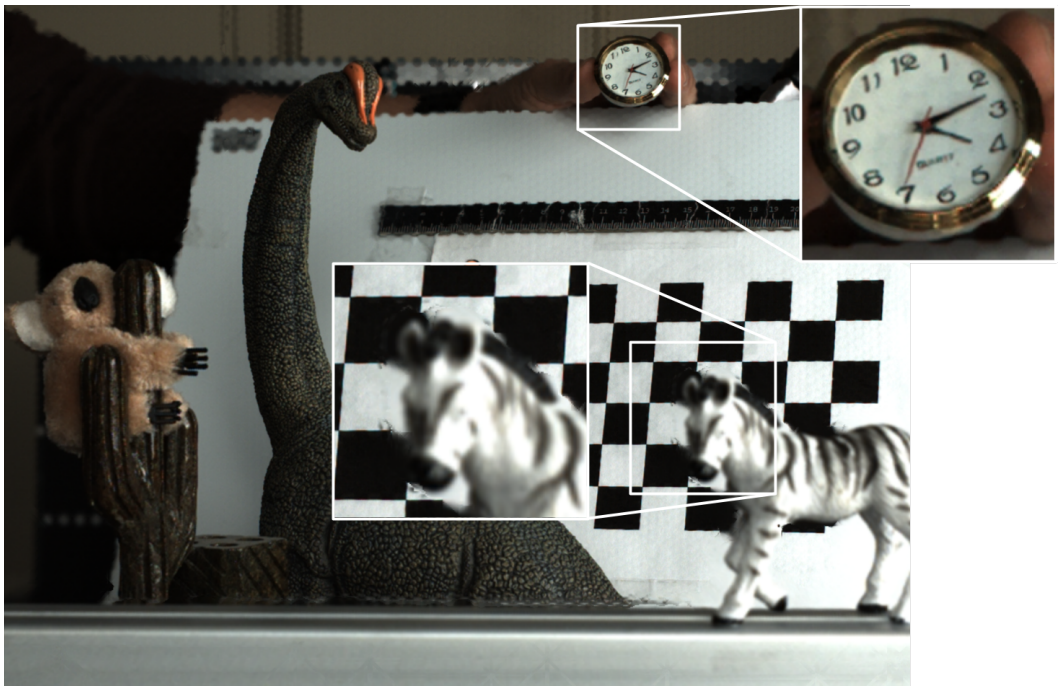


Figure 2.11: Example of post-refocusing with a plenoptic camera, at all distances, generating a so-called all in-focus image. Corresponding raw-image is given in Figure 2.5.

using a generative adversarial network (GAN), and leveraging the multi-view SAIs representation to compress the light-field. For more details, the reader can refer to the review of Monteiro and Nunes [78].

2.4.2 Applications in robotics

Light-field microscopy. An important application of plenoptic camera in this field has been demonstrated by Levoy *et al.* [7], [79]. Mignard-Debise and Ihrke [80] explored the use of consumer light-field camera technology for the purpose of light-field microscopy. A review of light-field microscopy can be found in [81].

Light-field particle image velocimetry (PIV). Plenoptic camera performances have been studied for single-camera volumetric velocity measurement technique. Preliminary results on plenoptic particle image velocimetry (PIV) are presented in [82]. La Foy and Vlachos [83] presented a multiple plenoptic cameras reconstruction algorithm for PIV aiming at overcoming the trade-off between the spatial and angular resolutions. Fahringer and Thurow [84] proposed an algorithm based on computational refocusing with the addition of a post reconstruction filter to remove the out-of-focus particles, as well as a comparison of algorithms. Shi *et al.* [85] investigated the design of plenoptic camera for such applications. They found that the micro-lenses' geometry is the vital parameter that affects the overall system performance.

Robustness to adversarial conditions. Using a multi-focus plenoptic camera, Nonn *et al.* [86] applied light-field PIV for metrology of spray droplets, measuring their size and velocity. More recently, Hasirlioglu *et al.* [87] investigated the potential of plenoptic cameras in the field of automotive safety, especially in adverse weather conditions. Their initial results show that raindrops cause deviations in depth values but can be corrected by the plenoptic camera. Similarly, Wu and Liu [88] used this observation for removing snowflakes from light-field image, based on deep learning methods. Yang *et al.* [89] aimed at removing raindrops from light-field images. The original image with raindrops is improved by refocusing on the far regions and filtering with a high-pass filter. Skinner and Johnson-Roberson [90] investigated the use of plenoptic camera for underwater 3D reconstruction where light attenuation and light scattering violate the brightness constancy constraint.

Objects detection. Applications to face detection and surveillance have been proposed and studied in [91]–[94]. Objects detection problem has been tackled in [95], [96]. Specific works aim at leveraging the light-field with transparent objects

such as [97]–[101]. Kaveti *et al.* [102] proposed to detect dynamic objects from the light-field and to remove them for static scene reconstruction.

Odometry. Taking inspiration from bio-compound-eyes, Neumann *et al.* [103] established the formalism for the plenoptic-based motion estimation. During his thesis, Dansereau [6] used the plenoptic function to achieve real-time navigation, introducing three distinct closed-form solutions to extract the motions parameters from the plenoptic function. At the same period, Dong *et al.* [8] gave a complete scheme to design usable real-time plenoptic cameras for mobile robotics applications. Johannsen *et al.* [104] introduced a novel Structure-from-Motion (SfM) pipeline based on Plücker ray coordinates. They deduced a set of linear constraints on ray space correspondences between a pair of light-field cameras. Ray space features have also be studied by Zhang *et al.* [105] for plenoptic SfM. Recently, Nousias *et al.* [106] presented a large-scale SfM pipeline tailored to light-field images, in which the scene is incrementally reconstructed. They later improved their framework, replacing the pose estimation by their linear approach to absolute pose estimation of [107]. Zeller *et al.* [108] adapted a SLAM formulation to deal with plenoptic information. Derived from their calibration model, they proposed a visual odometry framework [109], later improved with scale information [110]. Scene flow estimation have been studied by David *et al.* [111], using a local 4D affine model from sparse light-field that takes into account the epipolar structures. Tsai *et al.* [112] proposed the first derivation, implementation, and experimental validation of light-field image-based visual servoing.

Depth estimation. 3D reconstruction and/or depth estimation based on light-field imaging are one of the most important applications. An example of generated depth map can be seen in Figure 2.12. Plenoptic cameras allow to acquire passively a metric 3D representation of a scene in a single snapshot. For instance, Sardemann and Maas [113] analyzed depth accuracy and its variance for large distance 30 m to 100 m. They concluded that focused plenoptic camera is suited for applications in mobile robotics. Accuracy in the order of 3 % of the distance can be obtained for distance up to 100 m. This specific application will be analyzed in depth in chapter 4.

Figure 2.12: Example of a depth map obtained with a plenoptic camera by the built-in `Raytrix` software, corresponding to the raw-image given in Figure 2.5.

Conclusion

Several plenoptic imaging designs have been proposed to capture information that cannot be captured by conventional cameras from the plenoptic function. Such cameras capture only one point of view of a scene, whereas a *plenoptic camera* is a device that allows to retrieve *spatial* as well as *angular* information about a scene in a single exposure. In particular, this work focuses on plenoptic cameras based on a micro-lenses array (MLA) placed between a main lens and a sensor as illustrated in Figure 2.4. The specific design of such a camera allows to multiplex both types of information onto the sensor in the form of a micro-images array (MIA), as shown in Figure 2.5. The MLA position with respect to the main lens focal plane and the sensor plane determines the way the camera captures the light-field. It either corresponds to *unfocused* [4] or *focused* [5] configurations (see Figure 2.7).

The variety of applications of this type of sensor is great. For instance, this redundant information can be used for digitally refocusing and rendering [62] or for depth estimation [114]. Its capacities make the plenoptic cameras suited for applications in robotics. Calibration is an initial step for applications using plenoptic imaging. Precise and accurate camera parameters are usually required to obtain satisfactory results. In the next chapter, we answer the question "*How can we link world space information to the image space information?*", through the *calibration* problem of plenoptic cameras. Conventional cameras are usually represented using simple lens models. Due to the complexity of plenoptic cameras' design, the developed models are generally high dimensional. Specific calibration methods have to be proposed to retrieve the parameters of these models.

Chapter 3

Calibration of plenoptic cameras

Introduction

This chapter covers the problem of plenoptic cameras calibration. We first present the theoretical foundations of modeling optics elements, their properties, and how they relate to the calibration problem. In a second time, we review the existing solutions. We will see that current calibration methods rely on simplified projection models, use features from reconstructed images, or require distinct calibrations for each type of micro-lens. It is not satisfactory, especially when dealing with the multi-focus plenoptic camera because several parameters should be shared. Finally, we will present our solution for the calibration of plenoptic cameras, and the evaluation of our method.

Contributions

To the best of our knowledge, we propose the first calibration method for multi-focus plenoptic cameras that takes into account all types of micro-lenses within a single process. In order to exploit all available information, we propose to explicitly include the defocus blur in a new camera model. Thus, we introduce a new BAP feature defined in raw image space that enables us to handle the multi-focus case. We present a new pre-calibration step using BAP features from white images to provide a robust initial estimation of camera parameters. We use our BAP features in a single optimization process that retrieves intrinsic and extrinsic parameters of a multi-focus plenoptic camera directly from raw images of a checkerboard target.

In addition, we present an ablation study of the camera parameters and comparisons with state-of-the-art calibration methods. Several camera setups have been tested to validate the generalization of our method, and a simulation setup is proposed to evaluate our method in the unfocused configuration. Moreover, we take advantage of our BAP features to develop a new relative blur calibration process to link the geometric blur to the physical blur, i.e., the circle of confusion (CoC) to the point-spread function (PSF). This allows us to fully take advantage of blur in image space. Finally, we propose to use the blur to characterize the plenoptic camera in terms of depth of field (DoF).

3.1 Background

This section provides the mathematical background to understand how we can model the imaging process of a camera, especially how we are able to relate the three-dimensional world information to the two-dimensional image information through projection. Some optics properties will be given to complete the capabilities of real cameras, especially regarding the modeling of blur within such cameras. Finally, we will address the calibration process, i.e., how we are able to retrieve the parameters of these models.

3.1.1 Lens projection model

Our goal is to develop methods for performing metric measurements from images. Images are acquired by cameras which mapped the three-dimensional world information into two-dimensional image. This mapping is related to the camera, and we need to define a model explaining this process. In photogrammetry, this mapping is usually expressed in terms of the collinearity equations, whereas in computer vision it is usually expressed (equivalently) as a linear mapping of homogeneous coordinates. We will address here only geometric model. In the following, we will use homogeneous coordinates. Let $\boldsymbol{p}_w = \begin{bmatrix} x_w & y_w & z_w & 1 \end{bmatrix}^\top$ represent a point in world space and $\boldsymbol{p} = \begin{bmatrix} u & v & 1 \end{bmatrix}^\top$ be its projection in image space. The main idea is to find an expression for a matrix $\boldsymbol{K}$ such that

$$\boldsymbol{p} = \boldsymbol{K}\boldsymbol{T}\boldsymbol{p}_w. \tag{3.1}$$

Intrinsic parameters refer to the parameters of the camera model corresponding to the matrix $\boldsymbol{K}$. It does not depend on the position and orientation of the camera in space, which are modeled by the *extrinsic* parameters, the matrix $\boldsymbol{T} \in \mathrm{SE}(3)$.

3.1.1.1 Pinhole model

First, let's introduces the most used and simplest geometric model corresponding to the pinhole camera (see Figure 3.1(a)). In this model, only the principal light ray is allowed to go through. The lens aperture is infinitesimal, i.e., reduced to a single point. It means that everything is supposed to be in focus, which is not the case with real lenses. The projection is given by

$$\boldsymbol{p} = \boldsymbol{K}_{\text{pinhole}}\boldsymbol{P}\boldsymbol{T}\boldsymbol{p}_w \Leftrightarrow \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \propto \begin{bmatrix} a & c & u_0 \\ 0 & b & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} {}^c\boldsymbol{R}_w & {}^c\boldsymbol{t}_w \\ \boldsymbol{0}_{1\times 3} & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}, \tag{3.2}$$

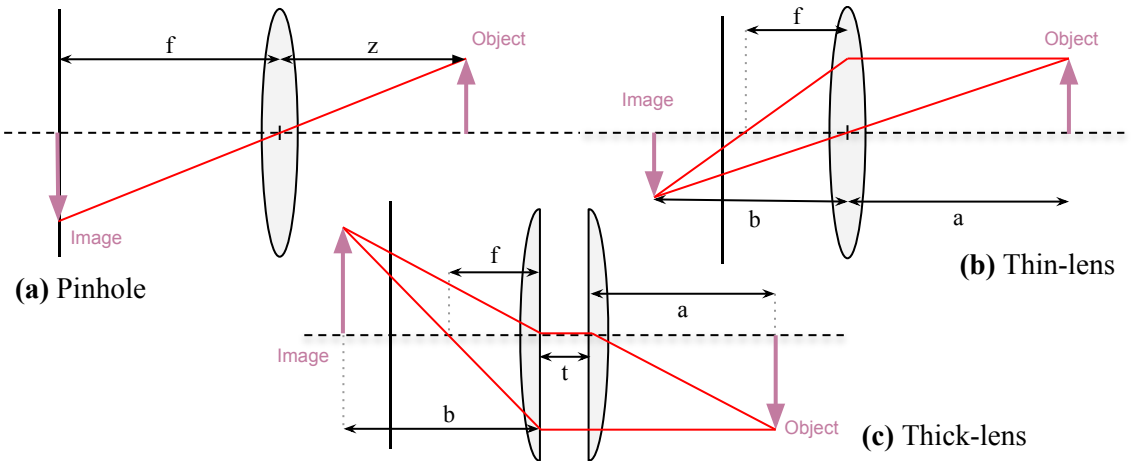


Figure 3.1: Examples of camera models.

where $\boldsymbol{K}_{\text{pinhole}}$ is the intrinsic matrix, $\boldsymbol{P}$ is the projection matrix, $\boldsymbol{T} \in \text{SE}(3)$ is the extrinsic matrix, i.e., the pose of the camera in world coordinates. $\begin{bmatrix} u_0 & v_0 \end{bmatrix}^\top$ is the principal point, i.e., the coordinates in pixel of the intersection between the sensor and the optical axis. The coefficients (a, b, c) are the scaling factors. The parameters (a, b) can be interpreted as the size of the focal length in horizontal and vertical pixels. The parameter c accounts for the pixel skew. In most cases, we consider the pixel squared, and we then have $a = b$ and $c = 0$.

3.1.1.2 Thin-lens model

To take into consideration the aperture of a real lens, and thus the other light rays within the cone of light, one of the most used models is the thin-lens model (see Figure 3.1(b)). The relationship between the focal length f, the object distance a and the image distance b, following the convention used in Table 3, is defined by the thin lens equation, also called the Gaussian lens equation, such as

$$\frac{1}{f} = \frac{1}{a} + \frac{1}{b}. \tag{3.3}$$

An alternate thin-lens equation (i.e., the Newton's form) can be derived as

$$\begin{cases} a = x - f \\ b = x' + f \end{cases} \implies xx' = -f^2. \tag{3.4}$$

If the sensor is placed at a distance equal to the focal length behind the lens, the objects infinitely far will be in-focus, and projected to a single point. A point that does not lie on the plane of focus is imaged to a disk on the sensor plane rather than a single point. The thin-lens equation is then used to project a point $\boldsymbol{p}_w$ into another three-dimensional virtual intermediate space behind the lens. Finally, the projection

is given by

$$\boldsymbol{p}' = \boldsymbol{K}_{\text{thin-lens}}\boldsymbol{T}\boldsymbol{p}_w \Leftrightarrow \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} \propto \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{f} & 1 \end{bmatrix} \begin{bmatrix} {}^c\boldsymbol{R}_w & {}^c\boldsymbol{t}_w \\ \boldsymbol{0}_{1\times 3} & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}, \tag{3.5}$$

where f is the focal length of the thin-lens, $\boldsymbol{K}_{\text{thin-lens}}$ is the intrinsic matrix, and $\boldsymbol{T} \in \text{SE}(3)$ is the extrinsic matrix.

3.1.1.3 Thick-lens model

The thin-lens model only holds for lenses whose thickness is negligible in comparison to the curvature of its faces. It can be extended to the thick-lens model (see Figure 3.1(c)) by introducing an offset t, i.e., the lens thickness, in the model, such that

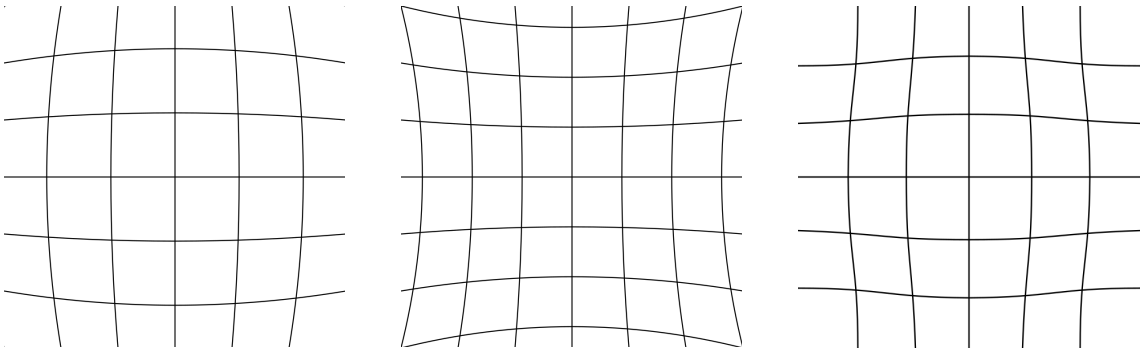
$$\boldsymbol{p}' = \boldsymbol{K}_{\text{thick-lens}}\boldsymbol{T}\boldsymbol{p}_w \Leftrightarrow \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} \propto \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & t \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{f} & 1 \end{bmatrix}}_{\boldsymbol{K}_{\text{thick-lens}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1-\frac{t}{f} & 0 \\ 0 & 0 & -\frac{1}{f} & 1 \end{bmatrix}} \begin{bmatrix} {}^c\boldsymbol{R}_w & {}^c\boldsymbol{t}_w \\ \boldsymbol{0}_{1\times 3} & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}, \tag{3.6}$$

where f is the focal length of the thin lens, $\boldsymbol{K}_{\text{thick-lens}}$ is the intrinsic matrix, and $\boldsymbol{T} \in \text{SE}(3)$ is the extrinsic matrix.

Survey on geometry of non-pinhole cameras can be found in [115]. Geometric models do not perfectly describe the physical projection. They are valid under the Gauss conditions, i.e., incidence angle near zero and rays close to the optical axis. Deviation from these hypotheses are usually taken into account in the distortion.

3.1.2 Distortion model

Distortion describes errors in the geometric projection through a lens, as illustrated in Figure 3.2. An undistorted point $\boldsymbol{p}_u = \begin{bmatrix} x_u & y_u & z_u & 1 \end{bmatrix}^\top$ expressed in the main lens frame after theoretical perfect projection (i.e., in the virtual intermediate space) can be distorted by applying a function φ to it, such as $\boldsymbol{p}_d = \varphi(\boldsymbol{p}_u) = \begin{bmatrix} x_d & y_d & z_d & 1 \end{bmatrix}^\top$.



(a) (b) (c)

Figure 3.2: Different types of distortion effect. (a) Barrel distortion, image magnification decreases with distance from the optical axis. (b) Pincushion distortion, image magnification increases with the distance from the optical axis. (c) Mustache distortion is a combination of both previous effects.

The process is modeled as

$$\begin{cases} x_d = \varphi^{(r)}(x_u) + \varphi^{(t)}(x_u) & = x_d^{(r)} + x_d^{(t)} \\ y_d = \varphi^{(r)}(y_u) + \varphi^{(t)}(y_u) & = y_d^{(r)} + y_d^{(t)} \\ z_d = \varphi^{(d)}(z_u) & = z_d^{(d)} \end{cases}, \tag{3.7}$$

with $\varphi^{(r)}$ being the radial distortion, $\varphi^{(t)}$ being the tangential distortion, and $\varphi^{(d)}$ being the depth distortion, defined in the following.

3.1.2.1 Lateral distortion

Distortion can follow many patterns. In general it is primarily radially symmetric but not always perfectly radially symmetric. The most used model is the distortion correction of Brown [116], also known as the Brown-Conrady model based on earlier work of Conrady [117]. It includes *radial* distortion, *tangential* distortion and the origin of distortion (i.e., decentered lens). Comparison of distortion models can be found in [118] and [119].

Decentering. The origin of distortion, defined as $\begin{bmatrix} o_x & o_y \end{bmatrix}^\top$ is not necessary the center of the image. We define the radius ς as the Euclidean distance to the origin of distortion as

$$\begin{cases} x_o = (x_u - o_x) \\ y_o = (y_u - o_y) \\ \varsigma = \sqrt{(x_u - o_x)^2 + (y_u - o_y)^2} = \sqrt{{x_o}^2 + {y_o}^2} \end{cases}. \tag{3.8}$$

Radial distortion. Radial distortion can be understood by its effect on concentric circles, as in an archery target. The radial component $\varphi^{(r)}$ of the distortion model is expressed as

$$\begin{cases} x_d^{(r)} = x_u + x_o \cdot (Q_0\varsigma^2 + Q_1\varsigma^4 + Q_2\varsigma^6 + \dots) \\ y_d^{(r)} = y_u + y_o \cdot (Q_0\varsigma^2 + Q_1\varsigma^4 + Q_2\varsigma^6 + \dots) \end{cases}, \tag{3.9}$$

with $\{Q_0, Q_1, Q_2, \dots\}$ describing the radial parameters. An alternative model exists for radial distortion known as the division model [120], [121].

Tangential distortion. With real lenses, real distortion is not necessary symmetric around a certain distortion center. Tangential distortion allows to account for this phenomenon. The tangential component $\varphi^{(t)}$ is expressed as

$$\begin{cases} x_d^{(t)} = \left(P_0(\varsigma^2 + 2{x_o}^2) + 2P_1 x_o y_o\right) \cdot \left(1 + P_2\varsigma^2 + P_3\varsigma^4 + \dots\right) \\ y_d^{(t)} = \left(P_1(\varsigma^2 + 2{y_o}^2) + 2P_0 x_o y_o\right) \cdot \left(1 + P_2\varsigma^2 + P_3\varsigma^4 + \dots\right) \end{cases} \tag{3.10}$$

with $\{P_0, P_1, P_2, P_3, \dots\}$ describing the tangential parameters.

3.1.2.2 Depth distortion

We can also include the depth distortion which is linked to Petzval field curvature (i.e., a slight change of focal length for points at greater distance from the optical axis), and influencing then only the z-depth component. Johannsen *et al.* [122, Eq. (5-6)] proposed a first model, expressed as

$$\begin{cases} \varsigma' = \varsigma \cdot (S_1 + S_2 z_d) \\ z_d^{(d)} = z_u + T_1\varsigma' + T_2{\varsigma'}^2 + T_3{\varsigma'}^4 \end{cases}. \tag{3.11}$$

Latter, they suggested that the influence of the distorted depth z is purely linear and that the distortion changes linearly with the depth. Therefore only one parameter D_0 is needed to model the relationship between the depth and the amount of depth distortion. In [123, Eq. (2.11)], the depth distortion is defined as

$$z_d^{(d)} = z_u + (1 + D_0 z_u)\left(D_1\varsigma^2 + D_2\varsigma^4 + \dots\right). \tag{3.12}$$

In case of the focused plenoptic camera and based on their depth calibration, Zeller *et al.* [124, Eq. (17)] defined the depth distortion as a function of the lateral distortion and thus, according to them, reflecting the physical reality. It is expressed as

$$z_d^{(d)} = z_u + \left(D_0 z_u\varsigma^2 + D_1{z_u}^3\varsigma\right). \tag{3.13}$$

Depth distortion has also been studied by Heinze *et al.* [125] and Zeller *et al.* [126]. But Zeller *et al.* [109] and Noury [127] both empirically observed that the effects of depth distortion, for large focal length and for large object distance, can be neglected compared to stochastic noise of the depth estimation process.

3.1.2.3 Inverse distortion

Once estimated, distortion coefficients are used to correct lens projection. Inverse distortion φ^{-1} is used to create an undistorted point $\boldsymbol{p}_u = \varphi^{-1}(\boldsymbol{p}_d) = \begin{bmatrix} x_u & y_u & z_u & 1 \end{bmatrix}^\top$ from a distorted point $\boldsymbol{p}_d = \begin{bmatrix} x_d & y_d & z_d & 1 \end{bmatrix}^\top$. The inversion of distortion model is not straightforward in the general case. For instance, the Brown-Conrady model is not invertible. Many methods have been proposed for this purpose including iterative techniques by Zhang [128] and Alvarez *et al.* [129], or also approximation techniques of Mallon and Whelan [130]. An efficient way to characterize the inverse distortion is to use a high order version of the Brown's model as shown in [131].

3.1.3 Optics properties

3.1.3.1 Magnification

The magnification is the process of enlarging the apparent size, not physical size, of something. This enlargement is quantified by a calculated number also called magnification. The linear magnification of an imaging system using a single lens is given by

$$\gamma = \frac{b}{a} = \frac{f}{f-a} = \frac{b-f}{f}, \tag{3.14}$$

where b (*resp.,* a) is the distance between the lens and the image (*resp.,* the object). The magnification can be positive or negative. If the image is real, i.e., $b > 0$, we have $\gamma > 0$, meaning that the image would be *upside-down*. A virtual image has $b < 0$, so γ is negative, meaning that the image is *upright*. The principle of magnification allows then to derive the distance between the lens and the imaging plane as

$$b = f\left(1 + |\gamma|\right). \tag{3.15}$$

3.1.3.2 F-number

The f-number of an optical system is the ratio of the system's focal length f to the diameter of the entrance pupil, A, such that

$$N = \frac{f}{A}. \tag{3.16}$$

The f-number accurately describes the light-gathering ability of a lens only for objects an infinite distance away. In optical design, an alternative is often needed for systems where the object is not far from the lens. In these cases the working f-number is used. The working f-number is defined as

$$N^* = \frac{1}{2\text{NA}} \approx \left(1 + |\gamma|\right) N, \tag{3.17}$$

Table 3.1: Conventional and calculated f-number full-stop series

AV	4	5	6	7	8	9
N (*indicated*)	4	5.6	8	11	16	22
N (*calculated*)	4.0	5.657...	8.0	11.31...	16.0	22.62...

where NA is the numerical aperture and γ is the magnification of the current focus setting. In case of a thin lens, let a be given by

$$\frac{1}{f} = \frac{1}{a} + \frac{1}{D},$$

where f is the focal length and D is the distance between the sensor and the lens. Then we can expressed the magnification as

$$\gamma = D/a,$$

and thus we can rearrange Eq. (3.16) and Eq. (3.17) to obtain the working f-number expressed as:

$$N^* = \left(1 + \frac{D}{a}\right)\frac{f}{A} = D\left(\frac{1}{D} + \frac{1}{a}\right)\frac{f}{A} = \frac{D}{A}. \tag{3.18}$$

Note that the standard full-stop f-number conventionally indicated on the lens differs from the real f-number calculated. Those values are summarized in the Table 3.1. Using the aperture value AV, the f-number N is given by

$$N = \sqrt{2^{\mathrm{AV}}}. \tag{3.19}$$

3.1.3.3 Circle of confusion

The circle of confusion (CoC) is an optical spot caused by a cone of light rays from a lens not coming to a perfect focus when imaging a point source. It is also known as disk of confusion, circle of indistinctness, *blur circle*, or blur spot. From similar triangle and from Eq. (3.3), the blur radius of a point in an image can be expressed as

$$\begin{cases} r = \dfrac{1}{2}A\left(\dfrac{d-b}{b}\right) = \dfrac{1}{2}A\left(d\left(\dfrac{1}{f} - \dfrac{1}{a}\right) - 1\right) = \dfrac{Ad}{2}\left(\dfrac{1}{f} - \dfrac{1}{a} - \dfrac{1}{d}\right) & \text{[metric]} \\ \rho = r/s & \text{[pixel]} \end{cases} \tag{3.20}$$

where ρ is the radius of blur in pixel, s is the pixel size expressed in mm/pixel, r is the radius of blur, d is the distance between the considered lens and the sensor, A is the diameter of the considered lens, f is the focal of the considered lens, a is the distance of the object from the lens, b is the distance of the image from the lens, all expressed in mm.

3.1.3.4 Point-spread function

In continuous domain, the blur can be expressed as the response of an imaging system to an out-of-focus point using the point-spread function (PSF). Let $\mathcal{I}(x, y)$ be the observed blurred image of an object at a constant distance. The image can be computed as the convolution of the PSF noted $\mathrm{h}(x, y)$, with the in-focus image, $\mathcal{I}^*(x, y)$, such that

$$\mathcal{I}(x, y) = \mathrm{h} * \mathcal{I}^*(x, y)\,, \tag{3.21}$$

where $*$ denotes the convolution operator. If the lens aperture is circular and the level of blur low, the PSF $\mathrm{h}(x, y)$ can be efficiently modeled by a two-dimensional Gaussian given by

$$\mathrm{h}(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right), \tag{3.22}$$

where the spread parameter σ is proportional to the blur circle radius ρ. Therefore, we can write

$$\sigma \propto \rho \Leftrightarrow \sigma = \kappa \cdot \rho \tag{3.23}$$

where κ is a camera constant that should be determined by calibration [132], [133]. The calibration of this coefficient will be addressed in section 3.3.5. Note that the spread parameter σ depends on the object distance a, i.e., blur and depth are linked.

3.1.3.5 Circle of least confusion

In practice using real lenses, light rays are not exactly focused to a perfect point, but have an intensity distribution which depends on the whole imaging system. This intensity distribution is usually modeled by the PSF. A fairly good approximation of the spatial extent of the PSF is given by the smallest diffraction-limited spot resolved by a camera in wave optics, i.e., the radius of the first null of the Airy disc, which is

$$r^* = 1.22 \cdot \nu \cdot N^*, \tag{3.24}$$

where ν is the wavelength of the light and N^* is the working f-number of the imaging system[1]. Therefore, combining the latter and Eq. (3.17), the image of an ideal point is a spot with radius

$$r^* = \frac{0.61 \cdot \nu}{\mathrm{NA}}, \tag{3.25}$$

where NA is the numerical aperture. In practice, if r^* is greater than the pixel size s, we define the effective minimum resolvable spot size r_0, also referred to as the circle of least confusion, as

$$r_0 = \max\left(r^*, s/2\right). \tag{3.26}$$

[1] The number 1.22 is an approximation for the Rayleigh criterion defining the minimum resolvable angle with a circular aperture, i.e., the first zero of the order-one Bessel function of the first kind $J_1(x)$ divided by π.

3.1.3.6 Depth of field

The depth of field (DoF) is the distance between the nearest and the farthest objects that are in acceptably sharp focus. *Acceptably sharp focus* is defined using the CoC maximal radius. The DoF is determined by the focal length f, the distance to object a, the acceptable circle of confusion size r_0, and the aperture A. The approximate depth of field can be given by

$$\text{DOF} \approx \frac{2a^2 N r_0}{f^2} \approx \frac{2a^2 r_0}{fA}. \tag{3.27}$$

3.1.3.7 Focus distance

The focus distance h can either be measured or read from the focus scale of the lens that is used. It can be defined as the sum of the image distance b and the object distance a and extends from the image plane to the the plane in object space which would be in focus. The relation $h = a + b$ stands. Manipulating this equation, the image plane distance, that we will note H in the following, can be calculated from the focus distance h and the focal length f, by

$$H = \left| \frac{h}{2} \left(1 - \sqrt{1 - 4\frac{f}{h}} \right) \right|. \tag{3.28}$$

3.1.4 Calibration

The practical problem of camera calibration has been present in computer vision since the early days of three-dimensional applications. Examples of robotics application include SfM, 3D reconstruction, visual odometry, mapping, localization, and SLAM. In order to get *metric* results with these applications, it is required to know precisely the parameters of the camera model which most accurately represent the system. Calibration is the process of estimating the intrinsic and extrinsic parameters of a camera. For instance, if we want to estimate the displacement between two frames based on features correspondences, a wrong value for the focal length will result in a scale error. Values given in datasheet and/or by the manufacturer are usually not representative of the reality and are too imprecise. Even with the best efforts, each lens fabrication and assembly process will result to a unique lens with specific intrinsic parameter. Consequently, we have to calibrate the system before using it in an application. We distinguish two kind of approaches to solve the calibration problem.

Auto-calibration or self-calibration, is the recovery of intrinsics and extrinsics from an unknown scene based on the observation that even for unknown motions in

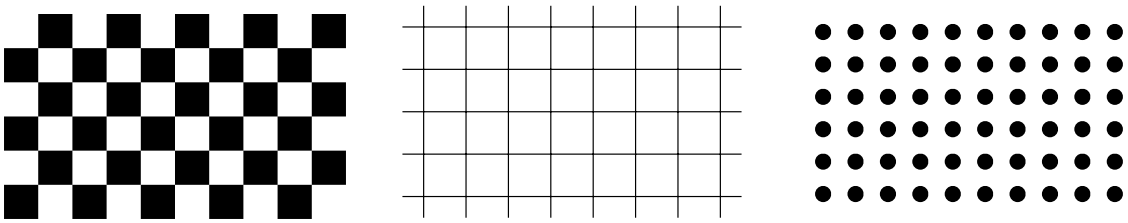
an unknown scene there are strong rigidity constraints relating the calibration to the images, scene and motion [134], [135].

Model-driven calibration is the recovery of intrinsic and extrinsic parameters from a scene where hypotheses of some states of the world are known and can be verified or confirmed by observing if the image projections conform to the hypotheses [136], [137]. Most approaches are based on calibration target with known parameters, for instance a checkerboard with know distance between corners, which projections, called features, are easily detectable in image space [128]. Correspondences between features and their counter-parts in 3D real world space allow to define criterion that can be used to solve or optimize the camera parameters.

Reviews and surveys on camera calibration can be found in [138]–[141]. Although standard for conventional cameras, existing methods are not easily translatable to plenoptic cameras. Due to the complexity of their design, the developed models are generally high dimensional. Specific calibration methods have to be proposed to retrieve the parameters of these models.

3.1.4.1 Calibration target

For model-driven camera calibration, usually a target with a known geometry is imaged. Some target implementations are illustrated in the Figure 3.3. There are also active targets such as target monitor [142]. The known pattern allows to extract feature points (e.g., corner, line, blob) from the image of the target. These points are reconstructed and put into correspondence with the known reference points on the target, i.e., the model. A cost function is evaluated according to a distance/residual between the model and the reconstructed points. By minimizing this function, the model is improved till the cost is low enough. A detailed description of a target based camera calibration algorithm is given by Tsai [136], with comments and improvements by Horn [137].



(a) (b) (c)

Figure 3.3: Different designs of calibration target. (a) Checkerboard target. (b) Line target. (c) circle pattern target.

Heinze [123], in his Master's thesis, worked on the design of the calibration target as it is the first step in implementing a calibration process. Pros and cons of each type have been analyzed and are resumed in [123, Table 3.1]. He concludes that the choice of a circle pattern target is favorable when using a light-field camera, especially for its robustness to noise. However, Dansereau [6] disagrees and suggests a classic checkerboard target to calibrate the camera. Indeed, center calculation from a circle pattern is prone to errors as the center of a projected ellipse is not always the center of the circle pattern when using a centroid method for instance. Furthermore, building the circle-based target model automatically means taking a new approach different to what is currently used in standard camera calibration techniques. In our work, we decided to use a standard checkerboard target.

3.2 Related work

In this section, we review the calibration models and methods from the literature, including, multi-cameras systems, *unfocused* plenoptic cameras, and *focused* plenoptic cameras. We do a focus on the special case of *multi-focus* plenoptic cameras. A non-exhaustive summary of calibration models is given in Table 3.2. We also briefly look at the calibration of the micro-lenses array (MLA). We tried to unify the notations with the ones used in this manuscript when possible to ease the comparison.

3.2.1 Multi-cameras calibration

First attend to calibrate sequences of un-calibrated hand-held camera for plenoptic modeling has been proposed by Koch *et al.* [143]. They used meshing constraints coupled with an existing SfM approach to calibrate a mesh of viewpoints. They relied on fused reconstructed depth maps to approximate the geometry.

Zhang [128] presented a full metric calibration of multi-cameras that computes intrinsic parameters and poses with respect to each position of the calibration grid for each camera independently. His method is largely used in the computer vision community as it is incorporated in the library `OpenCV`. It is considered as the standard method for camera calibration.

Vaish *et al.* [144] emphasized the fact that metric calibration is not necessary to calibrate an array of cameras. Their method can calibrate large arrays of discrete camera devices whose projection centers lie on the same plane. They used the hypothesis that cameras mostly lie on the same plane so they can then compute an affine transformation to align cameras in the same reference plane. They then

evaluated parallax for measurements in order to determine cameras position and are able to render the light-field given the computed depth.

Previous work has addressed calibration of collections of multiple cameras such as camera array, introducing more degrees of freedom in their model than are necessary to describe the models based on MLA. Georgiev *et al.* [145] show that the plenoptic camera is optically equivalent to an array of cameras, and the former is preferable due to the smaller size and the lower cost of a quality plenoptic implementation.

3.2.2 Unfocused plenoptic camera calibration

Dansereau *et al.* [146] introduced a ray model, drawing inspiration from [147], for the `Lytro` plenoptic camera [4]. They presented a 15-parameters model to decode the pixels into rays, including 10 parameters for the intrinsic matrix and 5 parameters for lateral distortion. They derived a camera rectification formulation that allowed a simple optimization algorithm for image calibration. Their intrinsic matrix $\boldsymbol{H} \in \mathbb{R}^{5\times5}$ allows to associate a ray $\boldsymbol{r}$ in light-field representation (2PP) to each decoded pixel $\boldsymbol{n}$ such as

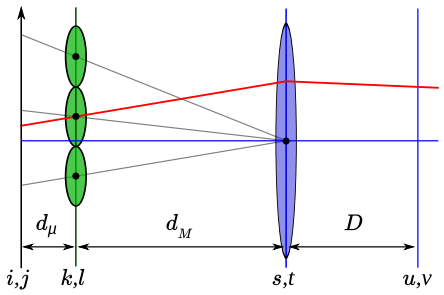


Figure 3.4: Dansereau *et al.* [146] camera model. Parameters equivalence: $D_M \to D$, and $d_\mu \to d$.

$$\boldsymbol{r} = \boldsymbol{H}\boldsymbol{n} \Leftrightarrow \begin{bmatrix} s \\ t \\ u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} H_{1,1} & 0 & H_{1,3} & 0 & H_{1,5} \\ 0 & H_{2,2} & 0 & H_{2,4} & H_{2,5} \\ H_{3,1} & 0 & H_{3,3} & 0 & H_{3,5} \\ 0 & H_{4,2} & 0 & H_{4,4} & H_{4,5} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} i \\ j \\ k \\ l \\ 1 \end{bmatrix}, \tag{3.29}$$

with (u, v, s, t) usual coordinates used in the two-parallel planes parametrization (2PP) [40], [41]; and (i, j) the pixel index within the (k, l) micro-image. Their optimization is based on ray reprojection objective function, which is the point-to-ray distance between the ray and the feature location. However, their model is not directly associated with physical parameters of the camera as they simply estimate values from the single matrix $\boldsymbol{H}$. Their algorithm is based on corner detection in SAIs, therefore not well suited to focused plenoptic cameras as SAI reconstruction is not an easy problem for this kind of configuration. Their method is publicly available as a `MATLAB` ToolBox at `https://github.com/doda42/LFToolbox`. Features include loading, visualizing and filtering light-fields, and, decoding, calibration and rectification of lenslet-based imagery.

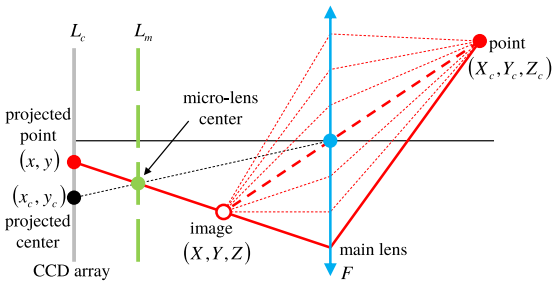


Figure 3.5: Bok *et al.* [148] camera model. Parameters equivalence: $L_m \rightarrow D$ and $L_c \rightarrow D{+}d$.

Bok *et al.* [148] formulated a geometric projection model to estimate intrinsic and extrinsic parameters from raw images directly (avoiding then the SAI reconstruction steps), including analytical solution and non-linear optimization. Their model includes two radial distortion parameters, non-skew pinhole parameters for the micro-lenses (i.e., f_x, f_y, c_x, c_y), and two coefficients K_1 and K_2 accounting for the main lens focal length (F), distances to the sensor ($D + d$), and to the MLA (D), with respect to the main lens, i.e.,

$$K_1 = \frac{(F - D) \cdot (D + d)}{F \cdot d} \qquad \text{and} \qquad K_2 = \frac{D \cdot (D + d)}{d}. \tag{3.30}$$

They introduced a novel line feature to overcome the difficulties in finding checkerboard corners and to improve the automation and accuracy of the feature identification. The linear features result from the interface of white and black squares. In [149], they extended their model with a one-to-one correspondence between pixels and rays, along with SAI generation procedure, and showed their method is applicable to the `Lytro Illum` camera. Although having less parameters than previous models, they do not take into account MLA misalignment with the sensor and the mean space between micro-lenses.

Liang and Ramamoorthi [150] derived an accurate model for light-field image formation by considering the full photo-sensor profile (inter-sensor distance, photo-sensor pitch size and angular sensitivity profile) and many physical parameters. They compared various lenslet-based cameras via simulation. Their model explains the success of the simple projection algorithm, i.e., its flexibility, and identified a few unique properties of the light-field camera in the inverse light transport analysis (i.e., spatially- and depth-variant details). They however evaluate their framework via simulation or only on `Lytro` light-field camera for real data.

Shi *et al.* [85] proposed a detailed model of a plenoptic camera in context of PIV. Based on linear optics, they derived a model based on ray-tracing: contrarily to previous method modeling the MLA as an array of pinholes, they modeled the main lens and each micro-lens as thin-lenses. As they are more interested in angular resolution, they restrained their camera prototype to act as the *unfocused* configuration, i.e., imposing the MLA to be positioned one focal length away from the sensor. They extended their work in [151] to develop a volumetric calibration method for plenoptic cameras still in context of PIV. Based on Gaussian optics, they relate a spatial voxel and its affected micro-lens and pixels through its circle of confusion (CoC) produced on the MLA plane. They take into account lens

defects and misalignment between MLA and image sensor by introducing five new parameters. The calibration method can calculate weighting coefficients for particle image reconstruction more accurately than the theoretical ray-tracing method. They model both the main lens and the micro-lenses as thin-lenses. Their calibration method is based on CoC and point-like feature based technique. They determined the blur circle diameter and center based on multiple observations of a point in micro-lenses, based on the following equation

$$r = -\alpha \frac{A}{2} \left(\gamma + \left(\frac{\beta x + \omega y + \varphi z + \delta}{S_i} - \frac{1}{\gamma} \right)^{-1} \right), \tag{3.31}$$

where r is the radius of the confusion circle, α is a correction factor to compensate ray prediction errors caused by optical aberrations and thin-lens model, $(\beta, \omega, \varphi, \delta)$ are coefficients to incorporate offset and rotation between the MLA and image sensor, γ is the magnification of the camera given by $\gamma = -S_i/S_o$, and S_o, S_i are the distances from the main lens to the focal plane and to the image plane (i.e., the distance between the main lens and the MLA, $S_i \to D$) respectively. Again, their method seems only suited for unfocused plenoptic camera as they assumed the distance between the MLA and the sensor to be equal to the focal length of the micro-lenses.

Hahne *et al.* [152] developed a ray model for the unfocused plenoptic camera by ray-tracing from the sensor side to the object space. They consider only the chief/principal ray, connecting micro-image centers (MICs) to the exit pupil center. They were the first to study how to apply triangulation to this kind of camera for depth estimation application.

Zhou *et al.* [153] proposed a practical two-step calibration method of lenslet-based unfocused light-field cameras. The calibration method describes the light-field camera parameters with specific physical meaning, related to the two planes parametrization (2PP). Their 8-intrinsic parameters model includes a simple distortion model, only composed of two coefficients for radial distortion. Their method is based on feature points extracted in SAI, allowing parameters to be retrieved using Zhang [128] method. First, the central SAI is used to estimate main lens parameters: the extrinsics, the radial distortion, the distance to the image plane (i.e., to the MLA, with $h'_m \to D$), and the principal point ($(x_0, y_0) \to (u_0, v_0)$). Second, feature points are extracted from all SAIs to estimate light-field disparity with the use of EPIs. Then, micro-lenses parameters, i.e., the sub-aperture size (D), the object distance (h_m), and the distance sensor-MLA ($b \to d$), are derived with line fitting method, as disparity is linked to those parameters. The micro-lens diameter is supposed known and used as "pixel size" for the virtual photo sensor. All planes are considered parallel, thus not taking into account MLA misalignment. Finally, they only compare themselves to the method of Dansereau *et al.* [146] and rely on SAI reconstruction software.

Bergamasco *et al.* [154] took a radically different path, focusing on the recovery of the geometry of generalized sensor rays in order to exploit them, whereas most of calibration methods are designed to be used in view rendering. Their non-parametric model [155] associates a ray for each pixel, modeled as $\boldsymbol{r}_i = \left[\boldsymbol{d}_i : \boldsymbol{p}_i\right]$ where $\boldsymbol{d}_i$ is the direction and $\boldsymbol{p}_i$ is the position of the ray in a reference frame. They analyzed the use of a calibration method that escapes the need to adopt a parametric model by exploiting dense correspondences generated using phase coding technique such as [156]. This is the first method attempting dense calibration with light-field cameras. It enables the adoption of a parameter-free optimization for non-central cameras, but their method works in the time rather than the space domain. The correct behavior is not guaranteed, mainly because of the sparsity of the MLA.

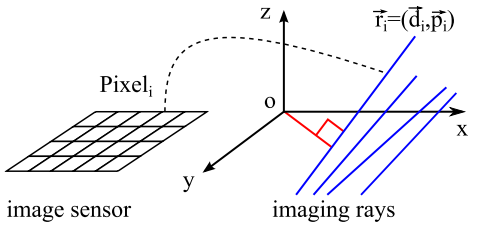


Figure 3.6: Bergamasco *et al.* [154] non-parametric camera model.

Hall *et al.* [157] built a third-order polynomial mapping functions to relate the real-world coordinates with light-field coordinates, in order to more accurately estimate three-dimensional geometric information from an in-house plenoptic camera. The mapping consists of 56 coefficients to be estimated. These light-field calibration methods are effective and particularly useful for three-dimensional geometry measurement. They restricted their work to the unfocused light-field camera.

In summary, most of the above methods require reconstructed images (SAIs) to extract features, and limit their model to the *unfocused* configuration, i.e., setting the sensor plane at the micro-lens focal plane. Therefore those models cannot be directly extended to the *focused* or *multi-focus* plenoptic camera.

3.2.3 Focused plenoptic camera calibration

With the arrival of commercial focused plenoptic cameras [5], [20], new calibration methods have been proposed. In this configuration, the micro-lenses focus on an intermediate image plane. We can distinguish two categories of methods: 1) the ones relying on reconstructed images, the SAIs; and 2) the ones operating directly on raw images.

3.2.3.1 Based on synthesized images

Johannsen *et al.* [122] formulated a general reprojection model in terms of the physical parameters of a `Raytrix` camera [5], [20], also using a 15-parameters model. They proposed a metric calibration and a distortion correction (lateral and depth) for multi-focus plenoptic cameras using a grid of circular patterns.

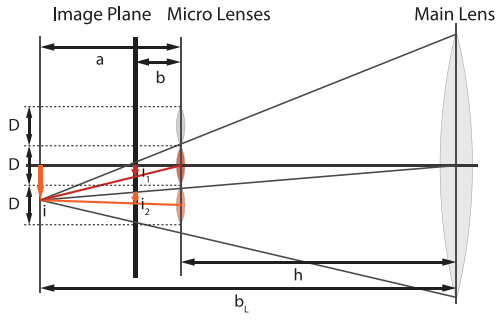


Figure 3.7: Johannsen *et al.* [122] camera model. Parameters equivalence: $D \rightarrow \Delta_\mu$, $b \rightarrow d$, and $h \rightarrow D$.

The estimated intrinsic parameters include the main lens focal length ($f \rightarrow F$), the focus distance, i.e., the distance between the MLA and the main lens ($h \rightarrow D$). The distortion consists of two coefficients for the offset ($(x_c, y_c) \rightarrow (o_x, o_y)$), five for the lateral, and five for the depth. The sequential quadratic programming algorithm was used to solve intrinsic and extrinsic parameters as well as the distortion coefficients. Their method required careful initialization of the optimization to converge due to high sensibility to local minima. No different micro-lens types were considered. It does not entirely model the camera and introduces then errors in depth reconstruction.

Heinze [123] implemented and tested an algorithm for metrically calibrating a focused plenoptic camera. For the calibration, an image of a circle pattern calibration target is taken with the `Raytrix` camera. The positions of features are extracted from the *all in-focus* reconstructed image. The points obtained this way are then projected through a camera model consisting of a thin-lens and array of pinholes, so that an error function can be minimized. However, the blob detection is done in the all in-focus image and the model relies on depth images obtained with the `RxLive` software of `Raytrix GmbH`. Heinze *et al.* [125] improved the previous work of Johannsen *et al.* [122]. They considered more sophisticated models of the main lens distortion by introducing a new parameter including its tilt/shift. They relate disparity map with metric space so that accurate 3D geometry measurement can be performed. They were able to differentiate each micro-lens type, calibrating then the distance between the MLA and the sensor $d^{(i)}$ for each type $i \in \{1, 2, 3\}$. The projection model and the metric calibration procedure are incorporated in the `RxLive` software of `Raytrix GmbH`.

Strobl and Lingenauber [158] presented a step-wise calibration approach to overcome the fragility in the initialization of the optimization. First, they determined the focal length ($f \rightarrow F$) and the radial distortion (including the offset and two coefficients). Second, they determined the internal offset of the MLA from the sensor ($b \rightarrow d$) and main lens ($h \rightarrow D$) respectively. However, they used all in-focus and virtual depth images in their calibration framework. They do not mention how they computed the all in-focus images, and rely on the `RxLive` software of `Raytrix GmbH`. Finally, the imaging process between the MLA and the sensor was not considered. Therefore, no geometrical parameters related to MLA could be calculated.

Zeller *et al.* [159] introduced two new methods to calibrate depth images obtained from focused plenoptic cameras, along with a method to calibrate the camera. The

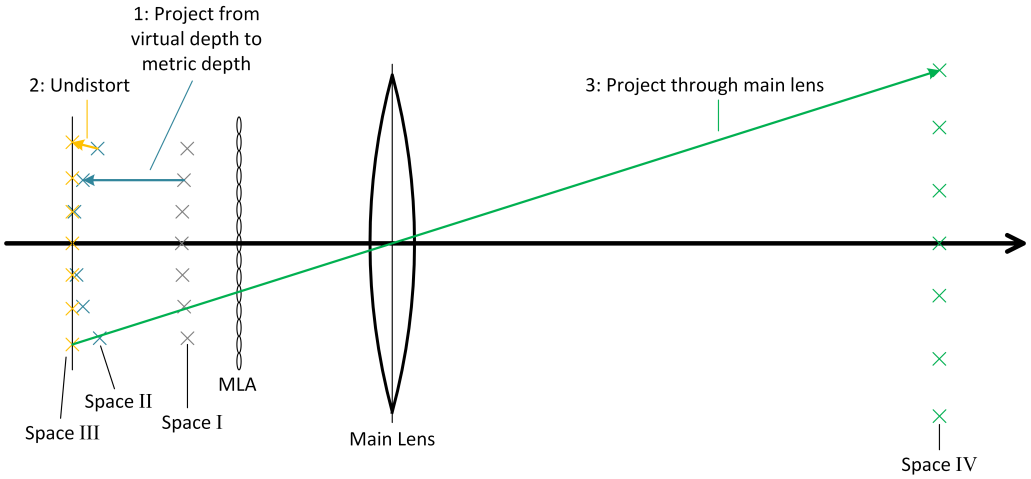


Figure 3.8: Heinze *et al.* [125] camera model. Virtual point is estimated, then reprojected into metric. Distortion is handled, and the point is projected through the main lens in object space.

MLA is assumed to be a pinholes grid which simplifies the path of rays. The calibration of the imaging process is based on Zhang [128] method applied on the central SAI. Contrarily to Johannsen *et al.* [122], they did not investigate the distortion of the depth map by the main lens, but proposed to use a separate optimization process for the depth calibration. To relate virtual depth to metric depth, they evaluated three models. In their first model, namely the *physical model*, they explicitly estimated the unknown parameters by fixing the focal length ($f_L \rightarrow F$). In their second model, namely the *behavioral model*, they estimated a linear combination of two measurable variables derived by rearranging some terms. Those methods are compared to the common *Curve Fitting* approach, where they estimate the coefficient of a polynomial function (i.e., Taylor-series), approximating the relation between the measured distance object and the virtual depth. All estimations are conducted using the least squared method. The two proposed model performed similarly, and both outperformed the curve fitting method.

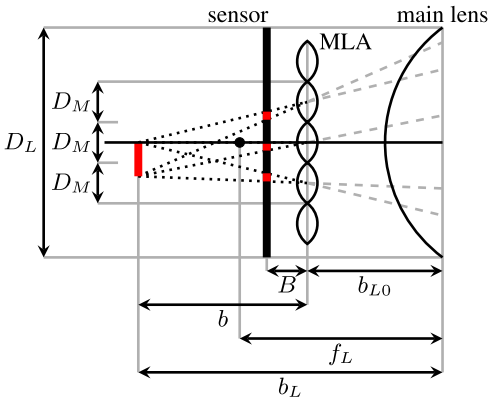


Figure 3.9: Zeller *et al.* [126] camera model. Parameters equivalence: $f_L \rightarrow F$, ($b_{L_0} \rightarrow D$, $B \rightarrow d$, $D_L \rightarrow A$, and $D_M \rightarrow \Delta_\mu$.

In [126] followed up by [124], they improved the camera projection model. They modeled the main lens as a thin-lens, whereas it was previously considered as a pinhole. The MLA is still modeled by an array of pinholes. The 5-intrinsic parameters composing

their model are: the main lens focal length ($f_L \rightarrow F$), the distance between the main lens and the MLA ($b_{L_0} \rightarrow D$), the distance between the MLA and the sensor, ($B \rightarrow d$), and the principal point, ($(c_x, c_y) \rightarrow (u_0, v_0)$). Their intrinsics also include the micro-lens centers, but are evaluated in a pre-calibration step, using white raw images. They considered a complete 7-parameters distortion model including lateral (i.e., radial and tangential) and depth distortion. They applied the distortion directly on light rays crossing the MLA thereby reflecting the physical reality, whereas in [122], [125], depth distortion is applied on virtual image point. Although they presented a complete distortion model (lateral and depth distortions), there is no consideration on the MLA misalignment nor its parameters (diameter, micro-lens types, etc.). The calibration process used the all in-focus image and virtual depth map to compute 3D observations.

O'Brien *et al.* [160] introduced a projection model used for their proposed calibration method. Their 7-intrinsic parameters model (including only one parameter for radial distortion) is composed of the main lens focal length F decomposed into (f^u, f^v) along the x- and y-axes, the principal point ($(c^u, c^v) \rightarrow (u_0, v_0)$), and, the two coefficients, (K_1, K_2), defined the same way as in Eq. (3.30). They presented a new feature called *plenoptic disc* (similar in nature to the CoC), defined by its center and its radius. Their feature parametrization is in 3D and is in one-to-one correspondence with point positions in the camera frame. They called the function that maps points to these features the *plenoptic projection*. They based their procedure on [149] adapted for their features. The features are detected in SAIs reconstructed from raw data. The minimization is conducted on their *plenoptic reprojection error* which is the distance between the plenoptic disc features and the expected features given estimated camera parameters. They compared themselves to Dansereau *et al.* [146], Bok *et al.* [149], and Nousias *et al.* [161]. To compensate lens aberrations, they only modeled radial distortion with a first order approximation. They assumed that the MLA is parallel to the main lens, so that all micro-lenses have a constant displacement. Thus, no misalignment is taken into consideration. The multi-focal arrangement would likely improve the feature-extraction process but was not considered here. This is the first method that successfully and reliably runs with both `Raytrix` and `Lytro` data with only minor pre-processing required.

All previous methods rely on reconstructed images (SAIs), which can lead to the introduction of errors in the reconstruction step as well as in the calibration process. However, computation of reconstructed images requires camera parameters and/or depth information to avoid artifacts and reconstruction error. To overcome this chicken and egg problem, several calibration methods focus on using only raw plenoptic images.

3.2.3.2 Based on raw images

Zhang *et al.* [162] proposed a calibration method using a parallel bi-planar checkerboard (i.e., to have a depth-scale prior) observations directly from raw images. They considered a detailed model of the MLA geometry that calibrates for non-planarity of the array. Their 10-parameters model includes the main lens focal length (F), the shift of image coordinates, i.e., the principal point (u_0, v_0), the main lens-MLA distance $(L \rightarrow D)$, the MLA-sensor distance $(l \rightarrow d)$, and the MLA misalignment, i.e., three rotations and two translations. The calibration is done in two steps, where all parameters except the main lens focal length are estimated first, and then F is estimated. During the optimization process, checkerboard planes are reconstructed in 3D space and the minimization is conducted on the distance between the computed plane inter-space and the ground truth. Note that in their calibration process the extrinsic parameters are not retrieved. They supposed that the micro-lens diameter Δ_μ is known, did not modeled the distortions caused by the lenses, and worked only on single focused plenoptic camera. In [163], they proposed a model based on the 2PP parametrization with 7-intrinsic parameters describing the 4D light-field and 8-distortion parameters. They simplified the focused plenoptic camera to be described by the 2PP. In their follow-up studies [164], they proposed a multi-projection-center model with 6-intrinsic parameters and 4-distortion parameters to describe light-field cameras also based on the 2PP. They proposed a calibration algorithm based on this model and on projective transformation, solved a close-form solution and a non-linear optimization by minimizing reprojection errors. The proposed model is applicable to both unfocused and focused plenoptic cameras.

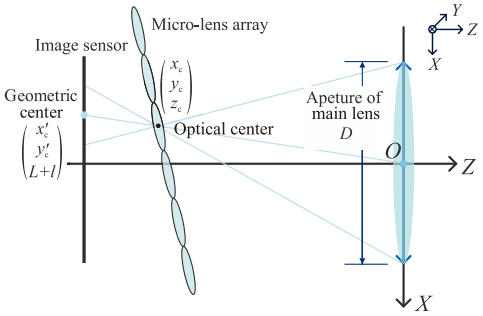


Figure 3.10: Zhang *et al.* [162] camera model. Parameters equivalence: $L \rightarrow D$, $l \rightarrow d$, and $D \rightarrow A$.

Sun *et al.* [165] proposed a calibration model in context 3D flame temperature measurement. They hand-determined the ratio of MLA distance to sensor with respect to MLA distance to the image plane for a specific point, allowing them to effectively identify the relative focal length of the main lens separately from the calibration process. They model both the main lens and the micro-lenses as pinholes. They adapted Zhang [128] calibration method to work with the focused plenoptic camera. Their model is composed of 5-intrinsic parameters, including the principal point of the main lens (u_0, v_0), the distance between the MLA and the sensor $(l_m \rightarrow d)$, the distance between the main lens and the sensor $(L \rightarrow D)$, and, the distance to the micro-lenses' image plane with respect to the MLA (S_v). They do not consider distortion models. They limited the extrinsic parameters estimation

to only one degree of freedom for the translation component, as the camera was assumed aligned with the flame to measure. Eventually, their pinholes assumption leads to a quite high reprojection error during the construction.

Nousias *et al.* [161] considered the geometric calibration of multi-focus plenoptic cameras in Galilean configuration. Their model is based on the one of Bok *et al.* [149] but operates on checkerboard corners, retrieved by a custom micro-image corner detector. Their method allows the detection of the type of micro-lenses, the retrieval of their spatial arrangement, and the estimation of intrinsic and extrinsic camera parameters, but for each one separately. Corners are reconstructed and the optimization minimizes the reprojection error in the raw plenoptic image. Micro-lenses identification is conducted only on micro-images containing corners. They computed the Tenenbaum Gradient (`Tenengrad`) score for each of this micro-images and classified them into $I = 3$ categories (with I, the number of micro-lenses type being made public). Then, to include micro-lenses type, they applied their method on each type of micro-lens independently. The model consists of the same 6-intrinsic parameters as in [149] without the distortion. The proposed method was appropriate for 3D reconstruction and structure-from-motion using multi-focus light-field cameras. However, they did not consider any distortion model, nor the potential MLA misalignment.

Noury *et al.* [166] presented a more complete geometrical model than previous work composed of 16-intrinsic parameters, including MLA misalignment, radial and tangential distortion. This model relates 3D points to their corresponding image projections, working directly with raw images. They developed a new detector to find checkerboard observations with sub-pixelic accuracy in each micro-image and use a pattern registration method to estimate their positions. The camera poses are then initially estimated using the Perspective-n-Point (PnP) algorithm [167], [168]. More details will be provided in section 3.3.3. Their 16-parameters model is composed of the main lens focal length ($f_L \rightarrow F$), the main lens distortion parameters (3 for radial and 2 for tangential), the distance between two micro-lenses centers ($d_\mu \rightarrow \Delta_\mu$), the MLA six degrees of freedom pose, to model the misalignment, and the sensor translations. There is no rotation considered as the sensor plane is supposed to be aligned with the main lens frame. They introduced a new cost function based on reprojection errors of both checkerboard corners and micro-lenses centers in the raw image space:

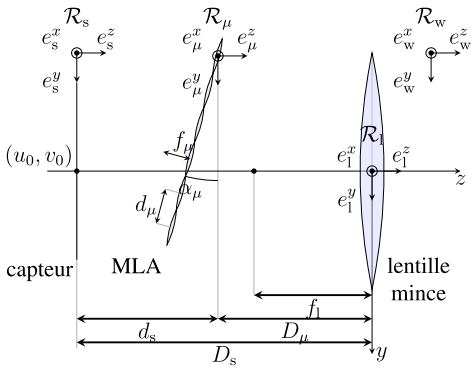


Figure 3.11: Noury *et al.* [166] camera model. Parameters equivalence: $f_L \rightarrow F$, $D_\mu \rightarrow D$, $d_s \rightarrow d$, and $d_\mu \rightarrow \Delta_\mu$.

- In the first term, for each frame, each checkerboard corner is reprojected into the image space through each micro-lens according to the projection model and compared to its observations.

- In the second term, the main lens center is reprojected according to a pinhole model in the image space through each micro-lens and compared to its detected micro-image center. It enforces projected micro-lens centers to get closer to their corresponding micro-image centers, making their method robust to wrong parameters initialization especially concerning those of the MLA.

However, they did not consider each micro-lens type and forced micro-lenses to act as pinholes. Although being able to detect checkerboard corners and borders, they did not exploit the latter in their optimization. Latter papers, [149], [161], [166], have achieved improved performance through automation and accurate identification of feature correspondences in raw images.

Wang *et al.* [169] proposed a geometrical calibration method for focused plenoptic camera based on virtual image points, establishing the mapping from object points to image points on the sensor after the main lens and the MLA. They suggested a forward model where: the mapping from object points to virtual image points is given by a pinhole model and the mapping from virtual image points to image points on detector is described by the second conjugate method, with micro-lenses considered as pinholes. Those two mapping parameters are estimated by detecting checkerboard corners in raw images and using Zhang [128] method. Each image point is associated to its corresponding micro-image centers through the inverse problem. They then used those parameters to calculate: the distance between the MLA and the sensor ($b \rightarrow d$), the distance between the MLA and the main lens ($L \rightarrow D$). Their method can be extended to calibrate multi-focus camera by considering each set of micro-lens types individually.

In conclusion, most of these methods rely on simplified models for optic elements: the MLA misalignment is not considered, and the micro-lenses are modeled as pinholes thus not modeling their apertures. Some do not consider distortions of the main lens or restrict themselves to the focused case. Finally, few have considered the multi-focus case [125], [149], [161], [169] but dealt with it in separate processes, leading to intrinsic and extrinsic parameters that vary depending on the type of micro-lens.

3.2.4 Micro-lens array calibration

Some methods only address on the calibration of the MLA, which is usually considered as a preliminary step for the calibration of the whole camera.

Cho *et al.* [170] introduced a method to calibrate micro-lens centers by searching for local maxima in the frequency domain over white images, that is images of uniform white targets. They estimate the MLA angle rotation given the selected frequency, and rotate the raw image. They then find center by eroding the image and paraboloid fitting to find precise local maximum. They refine the result by applying a Delaunay Triangulation procedure.

Thomason *et al.* [171] introduced a new method of calibration to estimate the position and orientation of the MLA. Micro-image centers are determined from calibration image obtained with a small aperture by calculating centroid with sub-pixel accuracy. The distance between the MLA and the main lens, D, is estimated using the principle of magnification, such as $D = F \cdot (\gamma + 1)$, where γ is the magnification and F the main lens focal length. Note that relation only stands in the unfocused case, i.e., the distance between the MLA and the sensor is equal to the focal length of the micro-lenses. The problem is solved by the Nelder–Mead simplex method to estimate the position and orientation of MLA as well as the distance between main lens and MLA. In this work, the directions of rays may deviate due to an inaccurate solution of the installation distances among main lens, MLA, and image sensor.

Xu *et al.* [172] proposed a robust estimator to accurately detect the micro-image centers. Based on that estimator, parameters that model the micro-images arrays are obtained by solving a global optimization problem. It includes skew along x- and y-axes, rotation angle and translation offset. Calibration is achieved in three steps: first, searching for the offset; second, optimizing the other parameters; and third, refining the offset.

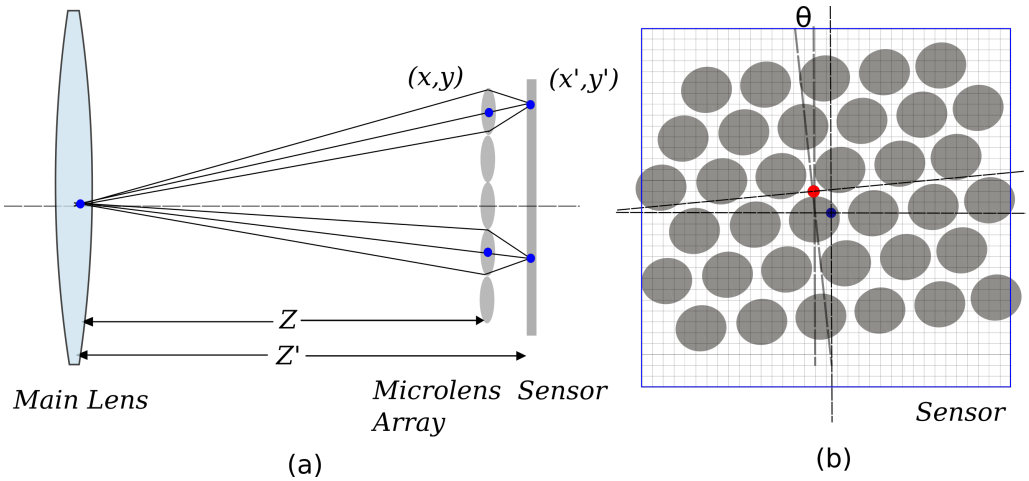


Figure 3.12: Xu *et al.* [172] micro-lens array model.

Suliga and Wrona [173] proposed a simple calibration of the MLA using raw images. This method allows to determine the micro-lenses pitch, the rotational offset, and the translation offsets. First, the raw image is demosaiced or debayered using

bilateral interpolation. Second, they gray-scaled the image as they do not need the color information, and they threshed the image. The detection of center position of each lenslet based on centroid method is performed on the thresholded image. They then made a bi-cubic interpolation over a 4×4 area to improve accuracy. Using the detected centers as input data, they determined the parameters using least-square estimation method.

The problem of the distortion caused by the MLA has been addressed by Li *et al.* [174]. They proposed a method for the local rectification of distorted images using white light-field images. The method consists of micro-lens center calibration, geometric rectification, and gray-scale rectification.

In our method, we will use a similar approach to Suliga and Wrona [173] to estimate the installation parameters of the MLA which will be refined later in a non-linear optimization process.

Table 3.2: Summary of calibration models from the literature

Reference	*Model*	*#Param.*	*Distortion*	*Objective Function*
Koch *et al.* [143]	sequences of camera	-	-	-
Zhang [128]	multi-camera	5+2	radial	residual reprojection
Vaish *et al.* [144]	array of cameras	relative pos.	-	rank-1 factorization
Dansereau *et al.* [146]	unfocused (Lytro)	10+5	lateral	point-to-ray distance
Bok *et al.* [148]	unfocused (Lytro)	6+2	radial	reprojected line quadratic distance
Zhou *et al.* [153]	unfocused (Lytro)	6+2	radial	residual reprojection + line fitting
Bergamasco *et al.* [154]	unfocused (in-house)	all rays	not directly	point-to-ray distance
Hall *et al.* [157]	unfocused (in-house)	poly. coeff.	not directly	least-square minimization
Johannsen *et al.* [122]	focused (Raytrix)	3+12	lateral + depth	quadratic residual reprojection
Zeller *et al.* [124]	focused (Raytrix)	5+7	lateral + depth	quadratic residual reprojection
Strobl and Lingenauber [158]	focused (Raytrix)	3+9	lateral + depth	quadratic residual reprojection
Heinze *et al.* [125]	multi-focus (Raytrix)	5+7	radial + depth	individual quadratic residual reproject.
Zhang *et al.* [162]	focused (in-house)	10	-	micro-lenses centers reprojection
Zhang *et al.* [163]	focused (in-house)	7+8	radial	residual reprojection
Bok *et al.* [149]	focused (Lytro Illum)	6+2	radial	reprojected line quadratic distance
Nousias *et al.* [161]	multi-focus (Raytrix)	6	-	individual quadratic residual reproject.
Noury *et al.* [166]	focused (Raytrix)	11+5	lateral	quadratic residual point reprojection + micro-lenses centers reprojection
O'Brien *et al.* [160]	focused (Raytrix + Lytro Illium)	6+1	radial	quadratic plenoptic reprojection
Labussière *et al.* [13]	multi-focus (Raytrix)	11+5+I	lateral	quadratic residual BAP reprojection + micro-lenses centers reprojection

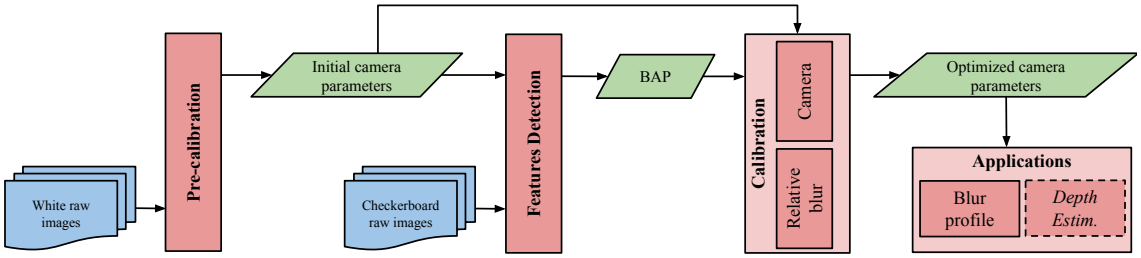


Figure 3.13: Overview of our proposed calibration method: first, the pre-calibration step retrieves initial camera parameters from white raw images at different apertures; then followed by the detection of BAP features that are used by the camera calibration process and calibration of the relative blur; finally, once the camera is calibrated, we show how to use our model to characterize the working range of the camera, or for metric depth estimation as will be presented in chapter 4.

3.3 Proposed calibration method (COMPOTE)

An overview of our method `COMPOTE` is given in Figure 3.13. This section is organized as follows. First, we present the camera model, its inversion, and how we model blur with our newly introduced BAP features. Second, we explain how we leverage raw white images in the proposed pre-calibration step to initialize camera parameters. Then, we detail the feature detection and the calibration processes, i.e., the camera calibration and the relative blur calibration.

3.3.1 Camera model

We consider the focused plenoptic camera, especially the multi-focus case as described by Perwaß and Wietzke [5] and Georgiev and Lumsdaine [25]. The camera is composed of a main lens and a photo-sensitive sensor with a micro-lenses array (MLA) in between, as illustrated in Figure 3.14. The micro-lenses array consists of I different types of lenses. The setup corresponds to the multi-focus system described by Perwaß and Wietzke [5] with $I = 3$. Note that our model can be applied to the single-focus plenoptic camera as well when $I = 1$. Finally, the unfocused configuration is a special case of our model where the micro-lens focal length is equal to the distance between the MLA and the sensor, i.e., $f = d$.

3.3.1.1 Camera configuration

When the camera is in the *unfocused* configuration, the distance separating the sensor and the MLA is equal to the focal length of the micro-lenses, i.e., $d = f$.

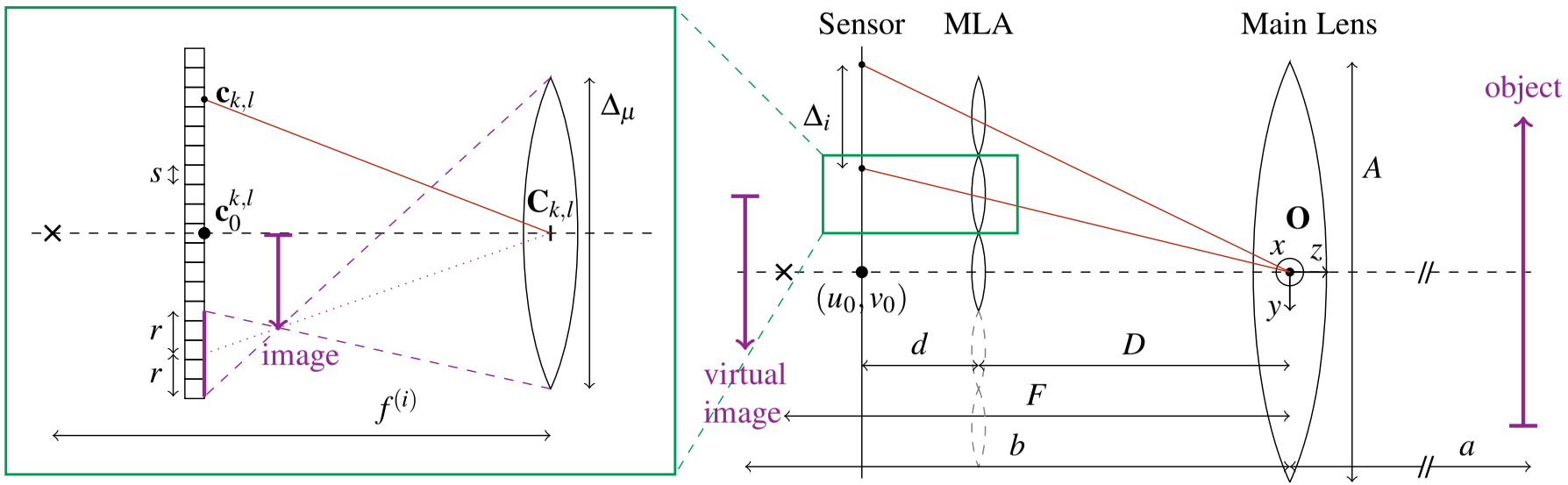


Figure 3.14: Focused plenoptic camera model with the notations used in this paper. Object points are projected by the main lens behind the micro-lenses array (MLA) into a virtual intermediate space, and then re-imaged by each micro-lens onto the sensor.

As seen in section 2.2.2, dealing with the focused plenoptic camera, we usually consider two possible configurations as presented by Georgiev and Lumsdaine [21]: 1) *Galilean*, when objects are projected behind the image sensor; and 2) *Keplerian*, when objects are projected in front of the image sensor. When considering micro-lenses as thin-lenses, we have to take into account their focal lengths to configure the camera. In practice, considering an object projected at distance b by the main lens, four cases are possible but only two are able to produce an exploitable image, i.e., with acceptable amount of blur, onto the sensor: $b < D$ and $f < d$ in Keplerian; and, $b > D$ and $f > d$ in Galilean. The condition $b > D$ can be achieved both when $F > D$ and $F < D$. The mode of operation is constrained by the focal length of the micro-lenses, as suggested by Mignard-Debise *et al.* [175]. We introduce the definition of the *internal* configuration according to the micro-lens focal length as

$$\begin{cases} f < d \implies & \text{Keplerian } \textit{internal} \text{ configuration,} \\ f > d \implies & \text{Galilean } \textit{internal} \text{ configuration.} \end{cases} \tag{3.32}$$

3.3.1.2 Main lens model

We model the main lens as a thin-lens and maps an object point to a virtual point in an intermediate space called the virtual space. An object at distance a is then projected at a distance b given the focal length F according to the thin-lens equation

$$\frac{1}{F} = \frac{1}{a} + \frac{1}{b}. \tag{3.33}$$

The main lens principal point is expressed as $\begin{bmatrix} u_0 & v_0 \end{bmatrix}^\top$ in image space. Yet with a plenoptic camera, the effect of the lens tilt can be directly observed as a tilt of the 3D image. This effect is known as the Scheimpflug principle [176]. However, this effect can also be compensated for by tangential distortion. Therefore, we chose not to include directly the tilt, and we model the main lens as parallel to the sensor plane. Furthermore, we define our camera reference frame as the main lens frame, with $\boldsymbol{O}$ being the origin, the z-axis coinciding with the optical axis and pointing outside the camera, and the y-axis pointing downwards. Distances are signed according to the convention summarized in Table 3: distances are positive when the point is real, and negative when virtual. Note that F is positive since our lens is convergent.

3.3.1.3 Direct distortion model

We consider distortion of the main lens. We model the radial and tangential components of the lateral distortion using the model of Brown-Conrady [116], [117] as seen in section 3.1.2. For direct projection, it corresponds to the u $\rightarrow$ d model. The image of a point $\boldsymbol{p}_u = \begin{bmatrix} x_u & y_u & z & 1 \end{bmatrix}^\top$ in the virtual intermediate space can be

distorted by applying a function φ to it, such that $\boldsymbol{p}_d = \varphi(\boldsymbol{p}_u) = \begin{bmatrix} x_d & y_d & z & 1 \end{bmatrix}^\top$. It is computed as

$$\begin{cases} x_d = x_u \left(1 + Q_1\varsigma^2 + Q_2\varsigma^4 + Q_3\varsigma^6\right) & \text{[radial]} \\ \quad + P_1\left(\varsigma^2 + 2{x_u}^2\right) + 2P_2 x_u y_u & \text{[tangential]} \\ y_d = y_u \left(1 + Q_1\varsigma^2 + Q_2\varsigma^4 + Q_3\varsigma^6\right) & \text{[radial]} \\ \quad + P_2\left(\varsigma^2 + 2{y_u}^2\right) + 2P_1 x_u y_u & \text{[tangential]} \end{cases} \tag{3.34}$$

where $\varsigma^2 = x_u^2 + y_u^2$. The three coefficients for the radial component are given by $\{Q_1, Q_2, Q_3\}$, and the two coefficients for the tangential by $\{P_1, P_2\}$.

Note that we do not include depth distortion in our model. Indeed, Zeller *et al.* [109] and Noury [127] both empirically observed that the effects of depth distortion, for large focal length and for large object distance, can be neglected compared to stochastic noise of the depth estimation process.

3.3.1.4 Inverse distortion model

Since our lateral distortion model is not invertible, we need to explicitly include the inverse distortion $\varphi^{-1}(\cdot)$ in the camera model. Inverse distortion is used to create an undistorted point $\boldsymbol{p}_u = \varphi^{-1}(\boldsymbol{p}_d) = \begin{bmatrix} x_u & y_u & z & 1 \end{bmatrix}^\top$ from a distorted point $\boldsymbol{p}_d = \begin{bmatrix} x_d & y_d & z & 1 \end{bmatrix}^\top$. For inverse projection, it corresponds to the d $\rightarrow$ u model. An efficient way to characterize the inverse distortions is to use a high order version of the Brown's model as shown in [131]. In particular, we use a model of the same order as our direct distortion model, and the mapping is expressed as

$$\begin{cases} x_u = x_d \left(1 + Q_{\text{-}1}\varsigma^2 + Q_{\text{-}2}\varsigma^4 + Q_{\text{-}3}\varsigma^6\right) & \text{[radial]} \\ \quad + P_{\text{-}1}\left(\varsigma^2 + 2x_d^2\right) + 2P_{\text{-}2} x_d y_d & \text{[tangential]} \\ y_u = y_d \left(1 + Q_{\text{-}1}\varsigma^2 + Q_{\text{-}2}\varsigma^4 + Q_{\text{-}3}\varsigma^6\right) & \text{[radial]} \\ \quad + P_{\text{-}2}\left(\varsigma^2 + 2y_d^2\right) + 2P_{\text{-}1} x_d y_d & \text{[tangential]} \end{cases} \tag{3.35}$$

where $\varsigma^2 = x_d^2 + y_d^2$. The three coefficients for the radial component are given by $\{Q_{\text{-}1}, Q_{\text{-}2}, Q_{\text{-}3}\}$, and the two coefficients for the tangential by $\{P_{\text{-}1}, P_{\text{-}2}\}$.

3.3.1.5 Micro-lenses array model

We also model the micro-lenses as thin-lenses allowing to take into account blur in the micro-images. The MLA consists then of I different lens types with focal lengths $f^{(i)}$ where $i \in [1\,..\,I]$ which are focused on I different planes. We make the hypothesis that all micro-lenses lie on the same plane. The MLA is approximately centered around the optic axis. We define the the origin of the MLA frame as the

center of the upper-left micro-lens. The coordinates axes are orientated the same way as the ones of the main lens. The structural organization of the lenses can be an orthogonal or hexagonal arrangement. Our model takes into account the MLA misalignment with the sensor, freeing its six degrees of freedom. The MLA origin is at a distance D from the main lens and at a distance d from the sensor.

Orthogonal approximation. We take into account the effect of the MLA tilt with respect to the sensor by using an orthogonal approximation since the angles are small. We thus consider specific distances $\mathrm{d}(k,l)$ (i.e., the distance between the micro-lens and the sensor) and $\mathrm{D}(k,l)$ (i.e., the distance between the micro-lens and the main lens) for each micro-lens (k,l). To ease the reading, we will only use the notation D and d in the following, but the quantities can be replaced by their corresponding orthogonal approximation.

Principal point. Furthermore, a detected micro-image center (MIC) usually does not coincide with the optical center of the considered micro-lens, as illustrated in Figure 3.14. We take into account this deviation in opposition to orthographic projection of MICs which causes inaccuracy in the decoded light-field. Therefore, the principal point $\boldsymbol{c}_0^{k,l}$ of the micro-lens indexed by (k,l) is given by

$$\boldsymbol{c}_0^{k,l} = \begin{bmatrix} u_0^{k,l} \\ v_0^{k,l} \end{bmatrix} = \frac{d}{D+d} \left(\begin{bmatrix} u_0 \\ v_0 \end{bmatrix} - \boldsymbol{c}_{k,l} \right) + \boldsymbol{c}_{k,l}, \tag{3.36}$$

where $\boldsymbol{c}_{k,l}$ is the center of the micro-image (k,l) expressed in pixel.

***f*-number matching principle.** The fundamental design principle for light-field imaging is that the working f-numbers of the micro-lenses and the main lens are matched. This condition maximizes the fill factor of the sensor while avoiding overlap between micro-images [4]. Both unfocused and focused plenoptic camera designs follow the f-number matching principle. As highlighted in [5], the micro images generated by the micro-lenses in a plenoptic camera should just touch to make the best use of the image sensor, meaning

$$\frac{d}{\Delta_\mu} = \frac{D_s - d}{A} \iff N_\mu^* = N^* - \frac{d}{A}, \tag{3.37}$$

where d is the distance between the MLA and the sensor, $D_s = D + d$ is the distance between the main lens and the sensor, Δ_μ and A are respectively the diameters of the micro-lenses and the main lens, and, N_μ^* and N^* are respectively the working f-numbers of the micro-lenses and the main lens.

Since typically $d \ll A$, we have $N_\mu^* \approx N^*$. So, the working f-numbers of the main imaging system and the micro lens imaging system should match. This also

implies that the design of the micro lenses fixes the f-number of the main lens that is used with the plenoptic camera.

3.3.1.6 Micro-images array model

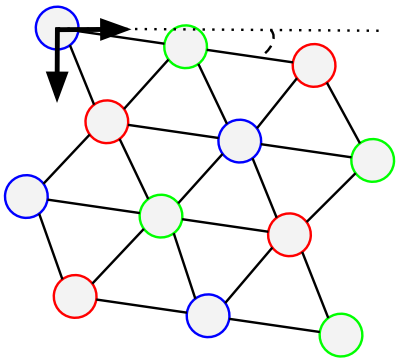


Figure 3.15: Illustration of the micro-images array (MIA) model.

Finally, each micro-lens produces a micro-image (MI) onto the sensor. The set of these micro-images has the same structural organization as the MLA. The data can therefore be interpreted as an array of micro-images, called by analogy the MIA. The MIA coordinates are expressed in image space. Let δ_i be the pixel distance between two arbitrary consecutive micro-images centers $\boldsymbol{c}_{k,l}$, i.e., the MIA pitch. With s the metric size of a pixel, let $\Delta_i = s \cdot \delta_i$ be its metric value, and Δ_μ be the metric distance between the two corresponding micro-lens centers $\boldsymbol{C}_{k,l}$, i.e., the MLA pitch. From similar triangles, we define the ratio λ between them by

$$\lambda \triangleq \frac{D}{D+d} = \frac{\Delta_\mu}{\Delta_i} \iff \Delta_\mu = \lambda \Delta_i = \frac{D}{D+d} \cdot \Delta_i. \tag{3.38}$$

We make the hypothesis that Δ_μ is equal to the micro-lens aperture. Finally, the MIA is characterized by its pitch δ_i, its pixel translation offset in image coordinates (τ_x, τ_y), and its rotation around the $(-z)$-axis, ϑ_z.

3.3.1.7 Defocus blur model

Defocus blur can be modeled by the CoC from a geometric point of view, or by the PSF from a physical point of view. As the micro-lenses act as a relay to re-image the projection of the main lens, we are interested in modeling blur generated by the micro-lenses. As they have a circular aperture of diameter Δ_μ, the blurred image is also circular in shape and is called the *blur circle*. From Eq. (3.20), the *signed* blur radius of the image of a point at a distance a from the micro-lens is expressed as

$$\begin{cases} r = \Delta_\mu \dfrac{d}{2} \left(\dfrac{1}{f^{(i)}} - \dfrac{1}{a} - \dfrac{1}{d} \right) & \text{[metric]} \\ \rho = r/s & \text{[pixel]} \end{cases} \tag{3.39}$$

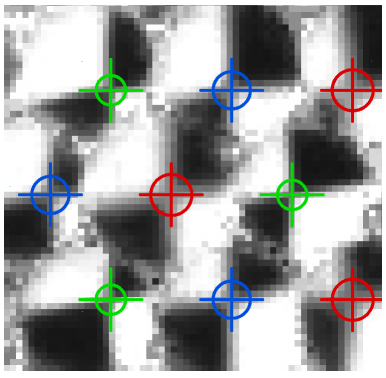


Figure 3.16: Illustration of our BAP features in raw images, described by their centers and their blur circles.

Each type of micro-lens will produce a specific blur radius as function of its focal length, providing us a way to distinguish them. To leverage blur information induced by the thin-lens model of the micro-lenses, we introduce a new blur aware plenoptic (BAP) feature characterized by its center and its radius, noted $\boldsymbol{p} = \begin{bmatrix} u & v & \rho & 1 \end{bmatrix}^\top$. The (u, v) part encodes the point position in the image and the ρ encodes the radius of the defocus blur. In other words, each micro-lens (k, l) projects virtual points onto the sensor at a position (u, v), with a blur radius ρ depending on the distance to the point and the micro-lens type. The BAP features are visualized in Figure 3.16.

Note that this is a geometric model. From a signal processing point of view, we prefer to model the process using the PSF as defined in Eq. (3.22). The spread parameter σ of the PSF is proportional to the blur circle radius ρ. It means that $\sigma = \kappa \cdot \rho$, and κ is a camera constant that we will determined by a specific calibration process described in section 3.3.5.

3.3.1.8 Direct projection model

Our complete plenoptic camera model links a scene point $\boldsymbol{p}_w = \begin{bmatrix} x & y & z & 1 \end{bmatrix}^\top$ to a BAP feature $\boldsymbol{p} = \begin{bmatrix} u & v & \rho & 1 \end{bmatrix}^\top$ in homogeneous coordinates through each micro-lens (k, l) of type (i). The direct projection $\Pi_{\mathrm{k,l}}$ is then given by

$$\begin{bmatrix} u \\ v \\ \rho \\ 1 \end{bmatrix} \propto \boldsymbol{\mathcal{P}}(i, k, l) \cdot \boldsymbol{T}_\mu(k, l) \cdot \varphi(\boldsymbol{K}_{\mathrm{thin-lens}}(F) \cdot \boldsymbol{T}_c \cdot \boldsymbol{p}_w)\,, \tag{3.40}$$

where $\boldsymbol{\mathcal{P}}(i, k, l)$ is the *blur aware plenoptic projection matrix* through the micro-lens (k, l) of type (i). $\boldsymbol{K}_{\mathrm{thin-lens}}(f)$ is the thin-lens projection matrix for the given focal length. $\boldsymbol{T}_c$ is the pose of the main lens with respect to the world frame and $\boldsymbol{T}_\mu(k, l)$ is the pose of the micro-lens (k, l) expressed in the camera frame. The function $\varphi(\cdot)$ models the lateral distortion. For $\boldsymbol{\mathcal{P}}(i, k, l)$, note that i can be function of the (k, l) indexes[2], and therefore not really an entry parameter. We chose to keep i in the

[2] In case of the `Raytrix` plenoptic camera, the number of micro-lenses types $I = 3$ is made public by the manufacturer, and the type (i) of the micro-image (k, l) is given by

$$i = ((l \mod 2) + k) \mod 3. \tag{3.41}$$

notation to ease the comprehension. It is computed as

$$\begin{aligned}\boldsymbol{\mathcal{P}}(i,k,l) &= \boldsymbol{P}(k,l)\cdot\boldsymbol{K}_{\text{thin-lens}}\big(f^{(i)}\big)\\ &= \begin{bmatrix} d/s & 0 & u_0^{k,l} & 0\\ 0 & d/s & v_0^{k,l} & 0\\ 0 & 0 & \Delta_\mu/2s & -\Delta_\mu d/2s\\ 0 & 0 & -1 & 0\end{bmatrix}\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & -1/f^{(i)} & 1\end{bmatrix}\\ &= \begin{bmatrix} d/s & 0 & u_0^{k,l} & 0\\ 0 & d/s & v_0^{k,l} & 0\\ 0 & 0 & \frac{\Delta_\mu d}{2s}\left(\frac{1}{d}-\frac{1}{f^{(i)}}\right) & -\frac{\Delta_\mu d}{2s}\\ 0 & 0 & -1 & 0\end{bmatrix}.\end{aligned} \tag{3.42}$$

The matrix $\boldsymbol{P}(k,l)$ projects the 3D virtual point onto the sensor and takes into account the blur radius. Note that we can recognize the blur radius formula applied to the micro-lens of aperture Δ_μ.

Finally, the direct projection model from Eq. (3.40) consists of a set Ξ of $(16+I)$ intrinsic parameters to be optimized, including:

- the main lens focal length F, expressed in $\boldsymbol{K}_{\text{thin-lens}}(F)$, and its five lateral distortion coefficients Q_1, Q_2, Q_3, P_1, and P_2, expressed in $\varphi(\cdot)$;
- the sensor translations, encoded in d and (u_0, v_0) through Eq. (3.36), from $\boldsymbol{P}(k,l)$;
- the MLA pose, including its three rotations $(\theta_x, \theta_y, \theta_z)$ and three translations (t_x, t_y, D), and the micro-lens pitch Δ_μ, expressed in $\boldsymbol{T}_\mu(k,l)$;
- and, the I micro-lens focal lengths $f^{(i)}$, in $\boldsymbol{K}(f^{(i)})$.

3.3.1.9 Inverse projection model

In order to relate image information to world information, we now inverse the projection model. We can back-project through a micro-lens (k,l) a BAP feature $\boldsymbol{p} = \begin{bmatrix}u & v & \rho & 1\end{bmatrix}^\top$ in homogeneous coordinates, at pixel (u,v) with a blur radius ρ, into a point $\boldsymbol{p}_w = \begin{bmatrix}x & y & z & 1\end{bmatrix}^\top$ in object space. The inverse projection $\Pi_{\text{k,l}}^{-1}$ is given by

$$\begin{bmatrix}x\\ y\\ z\\ 1\end{bmatrix} \propto \boldsymbol{T}_c^{-1}\cdot\boldsymbol{K}_{\text{thin-lens}}^{-1}(F)\cdot\varphi^{-1}\big(\boldsymbol{T}_\mu^{-1}(k,l)\cdot\boldsymbol{\mathcal{P}}^{-1}(i,k,l)\cdot\boldsymbol{p}\big). \tag{3.43}$$

Finally, the inverse projection model from Eq. (3.43) consists of a set Ξ' of 5-intrinsic parameters to be optimized, i.e., the inverse distortions coefficients $Q_{\text{-}1}$, $Q_{\text{-}2}$, $Q_{\text{-}3}$, $P_{\text{-}1}$, and $P_{\text{-}2}$ expressed in $\varphi^{-1}(\cdot)$.

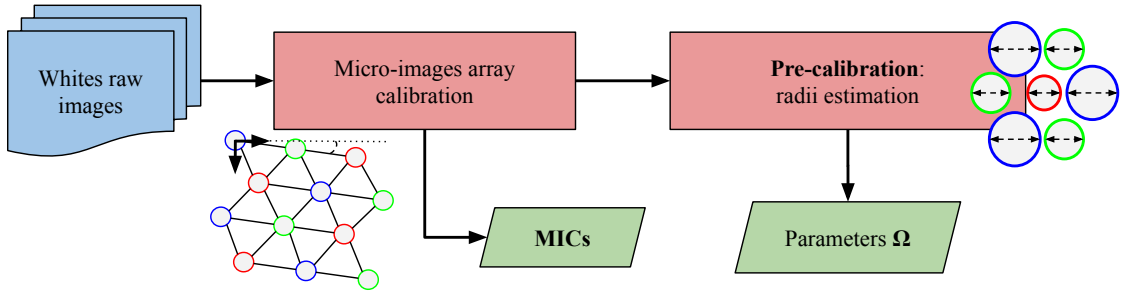


Figure 3.17: Overview of our pre-calibration step. From raw white images, the MIA is calibrated, and the MICs are extracted. Then internal parameters Ω are estimated based on micro-image radii measurements.

3.3.2 Pre-calibration using raw white images

The goal of the pre-calibration step is to provide a strong initial estimate of the camera parameters. It is illustrated in Figure 3.17. Inspired from *depth from defocus* theory [177], we leverage blur information to estimate our blur radius by varying the main lens aperture and using the different micro-lenses focal lengths, in combination with parameters from the image space. This is achieved by using raw white images acquired with a light diffuser mounted on the main objective, and taken at different apertures. Example of raw white image is given in Figure 3.18. We then show how the blur radii are linked to camera parameters, thus enabling their initialization.

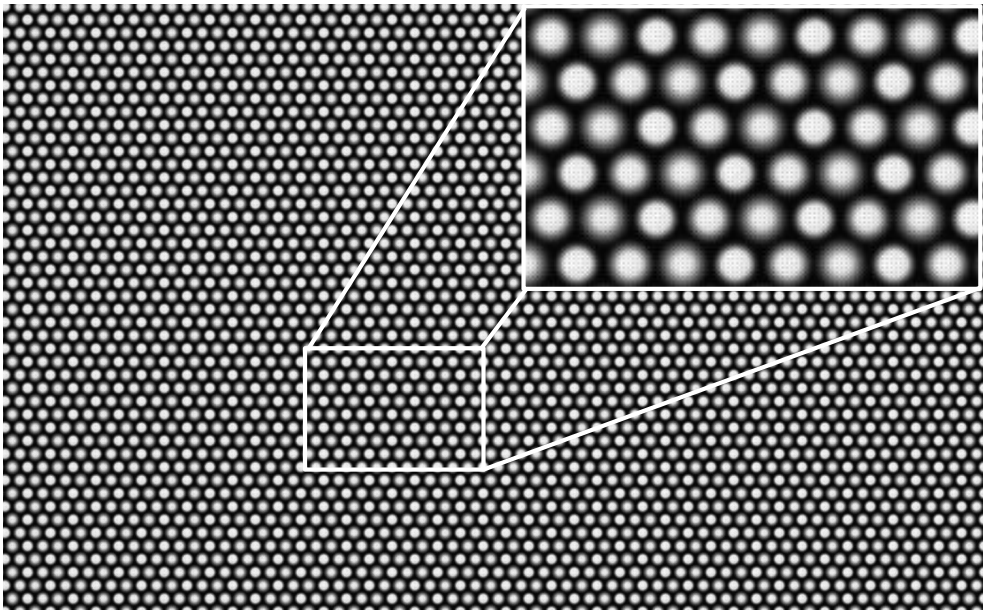


Figure 3.18: Example of raw white image taken with a `Raytrix R12` plenoptic camera, with the aperture set to $N = 4$.

3.3.2.1 Micro-images array calibration

First, the micro-images array (MIA) is calibrated using raw white images, based on the same process as in [127]. We compute the micro-image centers $\{\boldsymbol{c}_{k,l}\}$ by the intensity centroid method with sub-pixel accuracy [166], [173]. The distance between two micro-image centers, i.e., the MIA pitch δ_i, is then computed as the optimized edge-length of a fitted 2D regular grid mesh. 2D-2D correspondences are obtained by a nearest-neighbors search. Installation of the parameters is done by initializing the grid mesh with the 4-farthest micro-image centers as corners. A grid vertex $\boldsymbol{c}^*_{k,l} = \begin{bmatrix} u & v & 1 \end{bmatrix}^\top$ is computed depending on the structural organization of the MIA as

$$\begin{aligned}
&\text{Orthogonal:} && \begin{cases} u = \delta_i \cdot k \\ v = \delta_i \cdot l \end{cases} \\
&\text{Hexagonal (rows-aligned):} && \begin{cases} \tau_{\text{offset}} = \dfrac{1}{2} \quad \text{if } l \text{ is even, } \ 0 \text{ otherwise} \\ u = \delta_i \cdot (k + \tau_{\text{offset}}) \\ v = \delta_i \cdot l \cdot \sin \dfrac{\pi}{3} \end{cases} \\
&\text{Hexagonal (cols-aligned):} && \begin{cases} \tau_{\text{offset}} = -\dfrac{1}{2} \quad \text{if } k \text{ is odd, } \ 0 \text{ otherwise} \\ u = \delta_i \cdot k \cdot \cos \dfrac{\pi}{6} \\ v = \delta_i \cdot (l + \tau_{\text{offset}}) \end{cases}
\end{aligned} \tag{3.44}$$

Finally, the optimization is conducted by non-linear minimization of the distances between the grid vertices $\left\{\boldsymbol{c}^*_{k,l}\right\}$ and the corresponding detected MICs, i.e.,

$$\underset{\{\tau_x, \tau_y, \vartheta_z, \delta_i\}}{\arg\min} \sum_{(k,l)} \left\| \boldsymbol{T} \boldsymbol{c}^*_{k,l} - \boldsymbol{c}_{k,l} \right\|^2 \text{ with } \boldsymbol{T} = \begin{bmatrix} \cos\vartheta_z & -\sin\vartheta_z & \tau_x \\ \sin\vartheta_z & \cos\vartheta_z & \tau_y \\ 0 & 0 & 1 \end{bmatrix} . \tag{3.45}$$

The pixel translation offset in image coordinates, (τ_x, τ_y), and the rotation around the $(-z)$-axis, ϑ_z, are thus determined by the optimization process.

3.3.2.2 Deriving the micro-image radius

In white images taken with a light diffuser and a controlled aperture, each type of micro-lens produces a micro-image (MI) with a specific size and intensity. This provides a mean to distinguish between them (Figure 3.20). The process of capturing a white image is equivalent for the micro-lenses to imaging a white uniform object of diameter A at a distance D. The imaging process is schematized in Figure 3.19. Using optics geometry, the image of this object, i.e., the resulting micro-image, corresponds to the image of an "imaginary" point V constructed as the vertex of

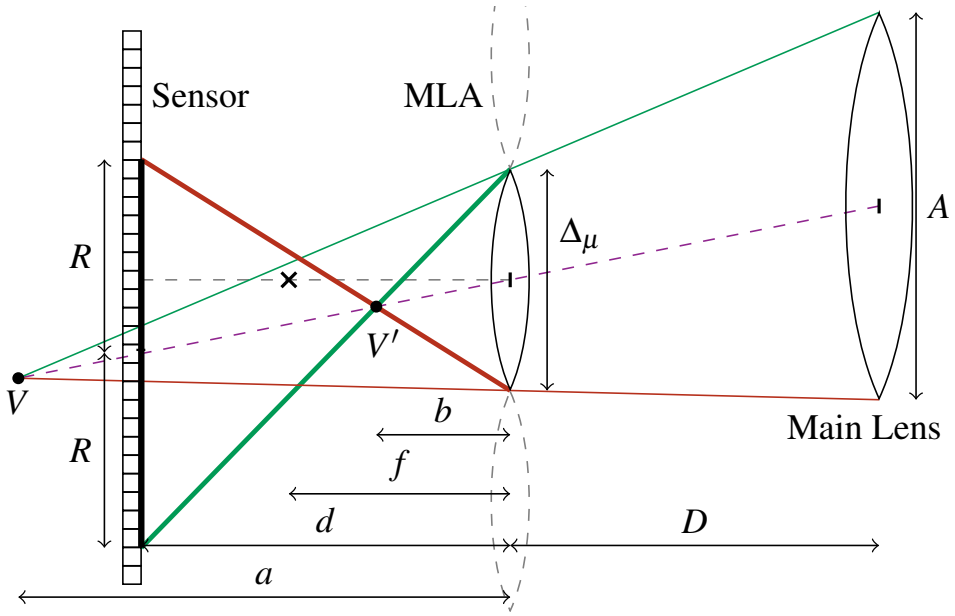


Figure 3.19: Formation of a micro-image with its radius R through a micro-lens while taking a white image using a light diffuser, at an aperture A, in Keplerian internal configuration. The point V is the vertex of the cone passing by the main lens and the considered micro-lens. V' is the image of V by the micro-lens and R is the radius of its blur circle.

the cone passing through the main lens and the considered micro-lens. Let a be the signed distance of this point from the MLA plane, expressed from similar triangles and Eq. (3.38) as

$$a = -D\frac{\Delta_\mu}{A - \Delta_\mu} = -D\left(A\left(\frac{D+d}{D}\cdot\frac{1}{\Delta_i}\right) - 1\right)^{-1}, \tag{3.46}$$

with A being the main lens aperture. Note the minus sign is added because the vertex is always formed behind the MLA plane, and thus considered as a virtual object for the micro-lenses. Geometrically, the micro-image (MI) formed is the *blur circle* of this "imaginary" point V. Therefore, injecting the latter expression in Eq. (3.39), the metric MI radius R is given by

$$\begin{aligned} R &= \frac{\Delta_\mu}{2} d\left(\frac{1}{f} - \frac{1}{a} - \frac{1}{d}\right) \\ &= \left(\Delta_i \cdot \frac{D}{D+d}\right) \cdot \frac{d}{2} \cdot \left(\frac{1}{f} + \left(A\left(\frac{D+d}{D}\cdot\frac{1}{\Delta_i}\right) - 1\right)\frac{1}{D} - \frac{1}{d}\right) \\ &= A \cdot \frac{d}{2D} + \left(\Delta_i \cdot \frac{D}{D+d}\right) \cdot \frac{d}{2} \cdot \left(\frac{1}{f} - \frac{1}{D} - \frac{1}{d}\right). \end{aligned} \tag{3.47}$$

From the above equation, the MI radius R depends linearly on the aperture of the main lens. However, the main lens aperture cannot be measured directly whereas we have access to the f-number value. Recall that the f-number of an optical system is

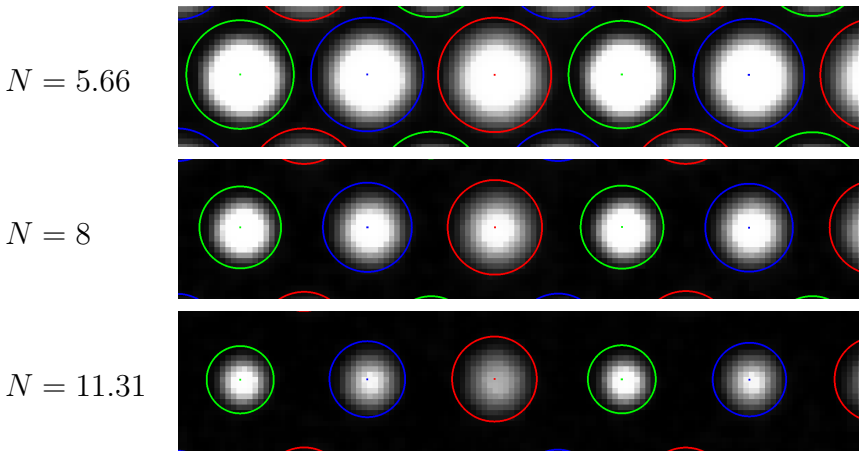


Figure 3.20: Micro-images from white raw images taken at different apertures. Each type of micro-lens is identified by its color (type (1) in *red*, type (2) in *green*, and type (3) in *blue*), and demonstrates a specific radius size.

the ratio of the system's focal length F to the aperture, A, given by $N = F/A$ (see Eq. (3.16)). Finally, we can express the MI radius for each micro-lens focal length type (i) as

$$\mathrm{R_i}\left(N^{-1}\right) = m \cdot N^{-1} + q_i \tag{3.48}$$

with

$$m = \frac{dF}{2D} \quad \text{and} \quad q_i = \frac{1}{f^{(i)}} \cdot \left(\Delta_i \cdot \frac{D}{D+d}\right) \cdot \frac{d}{2} - \frac{\Delta_i}{2}. \tag{3.49}$$

We thus relate the MI radius to the plenoptic camera parameters. It is a function of fixed parameters (d, D, F), measured parameters $(\Delta_i = s \cdot \delta_i)$ and variable parameters (N and $f^{(i)}$ with $i \in [1\,..\,I]$).

Let Ω be the set of parameters $\{m, q'_1, \ldots, q'_I\}$, where q'_i is the value obtained by

$$q'_i = \frac{1}{f^{(i)}} \cdot \left(\Delta_i \cdot \frac{D}{D+d}\right) \cdot \frac{d}{2} = q_i + \frac{\Delta_i}{2}. \tag{3.50}$$

They are used to compute the radius part of the BAP feature and to initialize the camera parameters.

Micro-image radii estimation. From raw white images, we measure each MI radius $\varrho = |R|\,/s$ in pixel based on image moments fitting. We use the second order central moments of the micro-image to construct a covariance matrix. The radius ϱ is proportional to the computed standard deviation Σ. Recall that raw moments and centroid of an image $\mathcal{I}(x, y)$ are given by

$$M_{ij} = \sum_{x,y} x^i y^j \mathcal{I}(x, y) \qquad \text{and} \qquad \{\bar{x},\ \bar{y}\} = \left\{\frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}}\right\}, \tag{3.51}$$

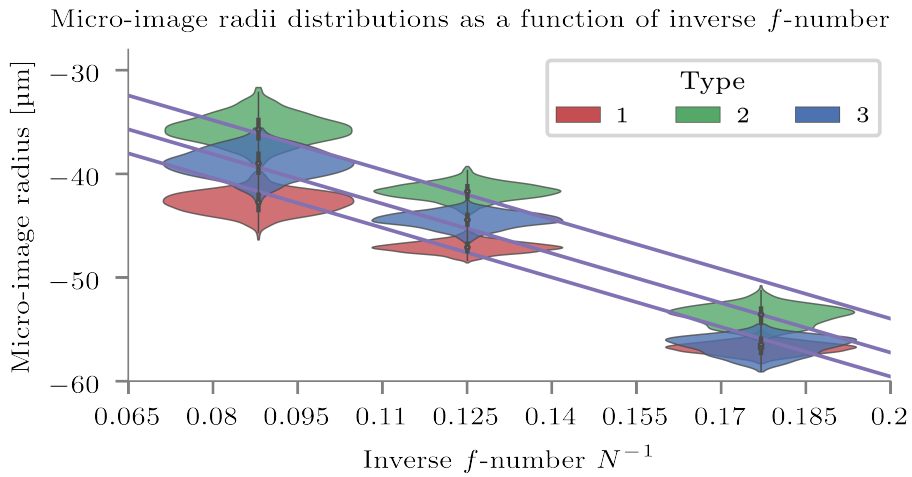


Figure 3.21: Micro-image radii as function of the inverse f-number (in *magenta*), with their distributions represented by the violin-boxes, for our camera consisting of $I = 3$ different types. Each type of micro-lens is identified by its color (type (1) in *red*, type (2) in *green*, and type (3) in *blue*) with its computed radius.

and the central moments by

$$\mu_{pq} = \sum_{x,y} (x - \bar{x})^p (y - \bar{y})^q \mathcal{I}(x, y)\,. \tag{3.52}$$

The covariance matrix is then computed as

$$\operatorname{cov}\left[\mathcal{I}(x, y)\right] = \frac{1}{\mu_{00}} \begin{bmatrix} \mu_{20} & \mu_{11} \\ \mu_{11} & \mu_{02} \end{bmatrix} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} \\ \sigma_{yx} & \sigma_{yy} \end{bmatrix}. \tag{3.53}$$

We define Σ as the square root of the greatest eigenvalue of the covariance matrix, i.e.,

$$\Sigma^2 = \frac{\sigma_{xx} + \sigma_{yy}}{2} + \frac{\sqrt{4\sigma_{xy}^2 + (\sigma_{xx} - \sigma_{yy})^2}}{2}. \tag{3.54}$$

The estimation is robust to noise, works under asymmetrical distribution and is easy to use, but requires a parameter α to convert the standard deviation Σ into a pixel radius $\varrho = \alpha \cdot \Sigma$. The parameter α is determined so that at least 98% of the distribution is taken into account. According to the standard normal distribution Z-score table, α is picked up in $[2.33, 2.37]$. In our experiments, we set $\alpha = 2.357$ as it best fits our measurements.

Recall that the pixel MI radius is given by $\varrho = |R| / s$. The metric radius is either positive if formed after the rays inversion, as in Figure 3.19, or negative if before, and thus depends on the *internal* configuration such as

$$R = \begin{cases} \varrho \cdot s & \text{[Keplerian } \textit{internal} \text{ configuration]}, \\ -\varrho \cdot s & \text{[Galilean } \textit{internal} \text{ configuration]}. \end{cases} \tag{3.55}$$

Coefficients estimation. Given several raw white images taken at different apertures, we estimate the parameters Ω, i.e., the coefficients of Eq. (3.48), for each type of micro-image. We use the f-number calculated from the aperture value AV by $N = \sqrt{2^{\mathrm{AV}}}$ (see Eq. (3.19)). The coefficient m is a function of fixed physical parameters independent of the micro-lenses focal lengths and the main lens aperture. Therefore, we obtain a set of linear equations, sharing the same slope, but with different y-intercepts. With $\boldsymbol{X} = \begin{bmatrix} m & q_1 & \dots & q_I \end{bmatrix}^\top$, the set of equations can be linearly rewritten as

$$\boldsymbol{A}\boldsymbol{X} = \boldsymbol{B}, \text{ and then } \boldsymbol{X} = \left(\boldsymbol{A}^\top \boldsymbol{A}\right)^{-1} \boldsymbol{A}^\top \boldsymbol{B}$$

where the matrix $\boldsymbol{A}$, containing the f-numbers and a selector of the corresponding y-intercept coefficient, and the vector $\boldsymbol{B}$, containing the radii measurements, are constructed by arranging the terms given the focal length at which they have been calculated. Finally, we compute $\boldsymbol{X}$ with a least-square estimation. Figure 3.20 shows examples of radii distributions from our experiments computed from white images taken at several f-numbers, and the estimated linear functions. In practice, at least two aperture configurations are required. More can be used to improve the estimation but on the condition that radii measurement distributions are distinguishable from each others, with small overlap.

3.3.2.3 Camera parameters initialization

First, the pixel size s is set according to the manufacturer values. The main lens focal length F is also initialized from them. Given the parameters Ω and the focus distance h, the parameters d and D are initialized as

$$d \longleftarrow \frac{2mH}{F + \xi \cdot 4m} \quad \text{and} \quad D \longleftarrow H - \xi \cdot 2d, \tag{3.56}$$

with $\xi = 1$ (*resp.,* $\xi = -1$) in Galilean (*resp.,* Keplerian) *internal* configuration, and where H is given by Eq. (3.28), i.e.,

$$H = \left| \frac{h}{2} \left(1 - \sqrt{1 - 4\frac{F}{h}} \right) \right|. \tag{3.57}$$

For completeness, note that the *unfocused* configuration can be initialized with $d \leftarrow 2m$ and $D \leftarrow F$.

In a second step, all distortion coefficients are initialized to zero. The principal point is set as the center of the image. The sensor plane is set parallel to the main lens plane, with no rotation, at a distance $-(D + d)$. Similarly, the MLA plane is initially set parallel to the main lens plane at a distance $-D$. From the pre-computed MIA parameters, the MLA translation takes into account the (x, y)-offsets $(-s\tau_x, -s\tau_y)$

and the rotation around the z-axis is initialized with $-\vartheta_z$. The micro-lenses pitch Δ_μ is set according to Eq. (3.38), where the ratio λ is computed using Eq. (3.56) such that

$$\lambda \longleftarrow \frac{F}{F + 2m}. \tag{3.58}$$

Finally, the initial micro-lenses' focal lengths are also computed from the parameters Ω as follows

$$f^{(i)} \longleftarrow \frac{d}{2 \cdot q'_i} \cdot \Delta_\mu. \tag{3.59}$$

Experiments will show that the initial model is close to the optimized model.

3.3.3 BAP features detection in raw images

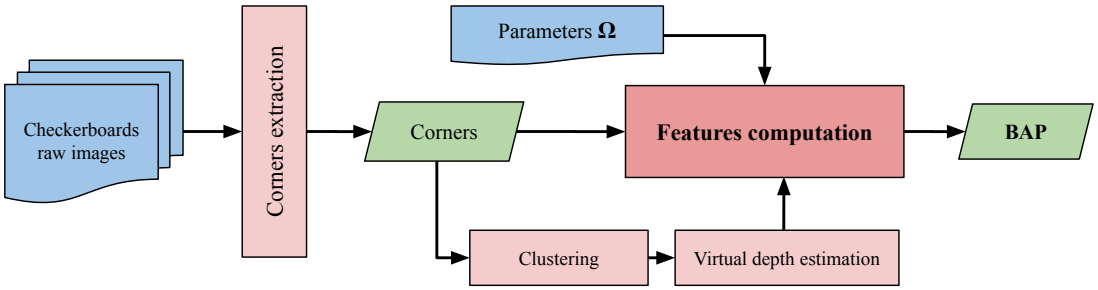


Figure 3.22: Overview of our BAP features detection step. From checkerboard raw images, corners are extracted and clustered. Virtual depth is then estimated and with the internal parameters Ω, the BAP features are computed.

At this point, the MIA is calibrated and MICs are extracted. The raw images are devignetted by dividing them by a white raw image taken with the same aperture. We based our method on a checkerboard calibration pattern. Indeed, corner-based calibration over-performs line-based calibration (as in [148]) as it enables the introduction of a 3D-to-2D reprojection error that is a representative performance measure of end-to-end imaging-system models [161]. The detection process, illustrated in Figure 3.22, is divided into two steps:

1. checkerboard images are processed to extract corners at position (u, v);
2. with the set of parameters Ω and the associated virtual depth estimate for each corner, the corresponding BAP feature is computed in image space.

3.3.3.1 Computing blur radius through micro-lens

To respect the f-number matching principle [5], we configure the main lens f-number such that the micro-images fully tile the sensor without overlap. In this configuration

the working f-number of the main imaging system and the micro-lens imaging system should match. We consider the general case of measuring an object $\boldsymbol{p}$ at a distance a from the main lens. First, $\boldsymbol{p}$ is projected through the main lens according to the thin lens equation,

$$\frac{1}{F} = \frac{1}{a} + \frac{1}{b}, \tag{3.60}$$

resulting in a point $\boldsymbol{p}'$ at a distance b behind the main lens, i.e., at a distance

$$a' = D - b, \tag{3.61}$$

from the MLA, as illustrated in Figure 3.14. From Eq. (3.20), the metric radius of the blur circle r of a point $\boldsymbol{p}'$ at distance a' through a micro-lens of type (i) is expressed as

$$\begin{aligned} r &= \left(\Delta_i \cdot \frac{D}{D+d}\right) \cdot \frac{d}{2} \cdot \left(\frac{1}{f^{(i)}} - \frac{1}{a'} - \frac{1}{d}\right) \\ &= \underbrace{\Delta_i \cdot \frac{D}{D+d} \cdot \frac{d}{2} \cdot \frac{1}{f^{(i)}}}_{=q_i' \text{ [Eq. (3.50)]}} - \underbrace{\Delta_i \cdot \frac{D}{D+d} \cdot \frac{d}{2} \cdot \frac{1}{d}}_{=\lambda\Delta_i/2 \text{ [Eq. (3.38)]}} - \underbrace{\Delta_i \cdot \frac{D}{D+d}}_{=\lambda\Delta_i \text{ [Eq. (3.38)]}} \cdot \frac{d}{2} \cdot \frac{1}{a'} \\ &= \left(-\lambda\Delta_i \cdot \frac{d}{2}\right) \cdot \frac{1}{a'} + \left(q_i' - \frac{\lambda\Delta_i}{2}\right). \end{aligned} \tag{3.62}$$

In practice, a' and d cannot be measured in raw image space, but the *virtual depth* can, as it will be shown in the next subsection. Virtual depth refers to a relative depth value. It is defined as the ratio between the signed object distance a' and the sensor distance d:

$$v = -\frac{a'}{d}. \tag{3.63}$$

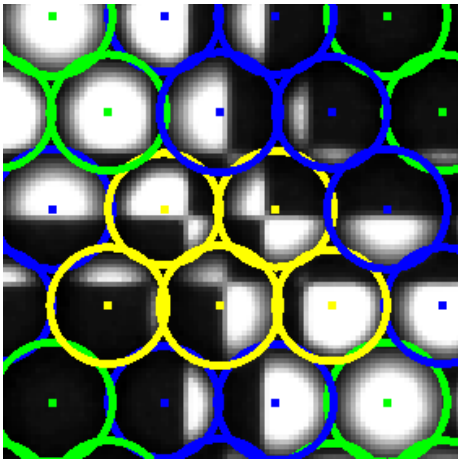
The sign convention is reversed for virtual depth computation. Distances are negative in front of the MLA plane. If we re-inject the virtual depth in Eq. (3.62), paying attention to the sign, and using Eq. (3.38), we can derive the radius of the *blur circle* of a point $\boldsymbol{p}'$ at a distance a' from the MLA by

$$r = \frac{\lambda\Delta_i}{2} \cdot v^{-1} + \left(q_i' - \frac{\lambda\Delta_i}{2}\right). \tag{3.64}$$

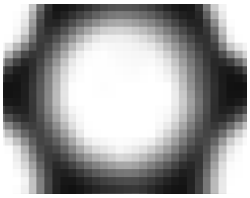
This equation allows to express the pixel radius of the blur circle $\rho = r/s$ associated to each point having a virtual depth directly in image space. It is done without explicitly evaluating the physical parameters A, D, d, F and $f^{(i)}$ of the camera, which allow us to introduce a new reprojection error.

3.3.3.2 Features extraction

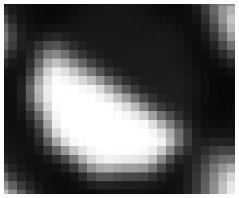
First, we detect corners in raw images using the detector introduced by Noury *et al.* [166] with sub-pixel accuracy in each micro-image. With a plenoptic camera, unlike



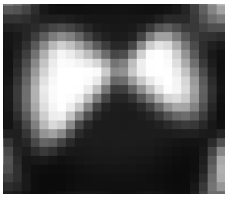
(a) `corner` (in yellow), `border` (in blue) and `full` (in green).



(b) `full` type



(c) `border` type



(d) `corner` type

Figure 3.23: Micro-images characterization of a checkerboard image given its content. Result of the characterization is illustrated by (a) and micro-image types by (b)-(d).

standard camera, a same point in object space is projected into multiple observations onto the sensor. The checkerboard is designed and positioned so that the sets of observations are sufficiently separated from each others to be clustered. We use the DBSCAN algorithm [178] to identify the clusters. We then associate each point with its cluster of observations. Secondly, once each cluster is identified, we compute the virtual depth v.

Corners extraction. The corner detector introduced in [166] is able to take into account multiple focal lengths using a scale parameter, where others may struggle with uncertainty due to blurred content. It classifies all micro-images according to their content type (`full`, `border` and `corner`, as illustrated by Figure 3.23) using histograms of gradients method.

Then, it detects corners with sub-pixelic precision in micro-image of type `corner` through an optimization process. To determine the position of the corner, it optimizes a warping function transforming a model corner image into the observed corner. The warping function has six degrees of freedom, including two translations, one rotation, one stretch, one shear, and one scale. The scale parameter controls the level of edge blurring through interpolation, allowing good fitting results and thus good corner localization in blurred micro-images.

Observations clustering. With plenoptic camera, unlike standard camera, an observed point is projected into more than one point on the sensor. Multiple observations correspond to the same observed point in the world. These observations are spatially close, and in the case of corners detected in checkerboards images, sets of observations are sufficiently far from each others to be clustered. Due to the nature

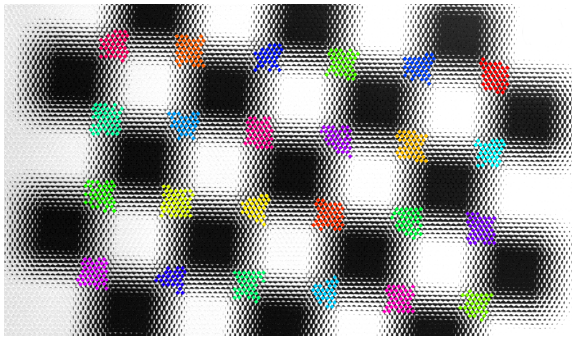


Figure 3.24: Clusters of observations in raw plenoptic image obtained by the DBSCAN algorithm.

of the spatial distribution of the data, we chose the density-based spatial clustering of applications with noise (DBSCAN) algorithm [178] to identify the clusters.

Virtual depth estimation. Once each cluster is identified, we can compute the virtual depth v from disparity δ. The disparity refers to the difference in coordinates of a same feature within two stereo images, here two micro-images. Let B be the distance between the centers of two micro-lenses $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$, i.e., the baseline. Let $\Delta\boldsymbol{p} = \|\boldsymbol{p}_1 - \boldsymbol{p}_2\|$ be the Euclidean distance between images of the same point in corresponding micro-images. The disparity is defined as

$$\delta = B - \Delta\boldsymbol{p}. \tag{3.65}$$

The virtual depth v is then calculated with the intercept theorem as

$$v = \frac{B}{\delta} = \frac{B}{B - \Delta\boldsymbol{p}} = \frac{\eta \cdot \Delta_\mu}{\eta \cdot \Delta_\mu - \Delta\boldsymbol{p}} = \frac{\eta \cdot \lambda\Delta_i}{\eta \cdot \lambda\Delta_i - \Delta\boldsymbol{p}}. \tag{3.66}$$

If we consider two adjacent micro-lenses, the baseline B is just the diameter of a micro-lens, i.e., $B = \Delta_\mu = \lambda\Delta_i$ and $\eta = 1$. For farther micro-lenses, the baseline is a multiple of that diameter, where η is not necessarily an integer. To handle noise in corner detection, we use a median estimator to compute the virtual depth of the cluster, taking into account all combinations of point pairs in the disparity estimation.

BAP features computation. Finally, we compute the BAP features from the blur radius formula Eq. (3.64), using the set of parameters $\Omega = \{m, q'_1, \ldots, q'_I\}$ and the available virtual depth v. In each frame n, for each micro-image (k, l) of type (i) containing a corner at position (u, v) in the image, the feature $\boldsymbol{p}^n_{k,l}$ is given by

$$\boldsymbol{p}^n_{k,l} = \begin{bmatrix} u & v & \rho & 1 \end{bmatrix}^\top, \text{ with } \rho = r/s. \tag{3.67}$$

In the end, our observations are composed of a set of micro-images centers $\{\boldsymbol{c}_{k,l}\}$ and a set of BAP features $\left\{\boldsymbol{p}^n_{k,l}\right\}$ allowing us to introduce two reprojection errors corresponding to each set of features as explain in the next section.

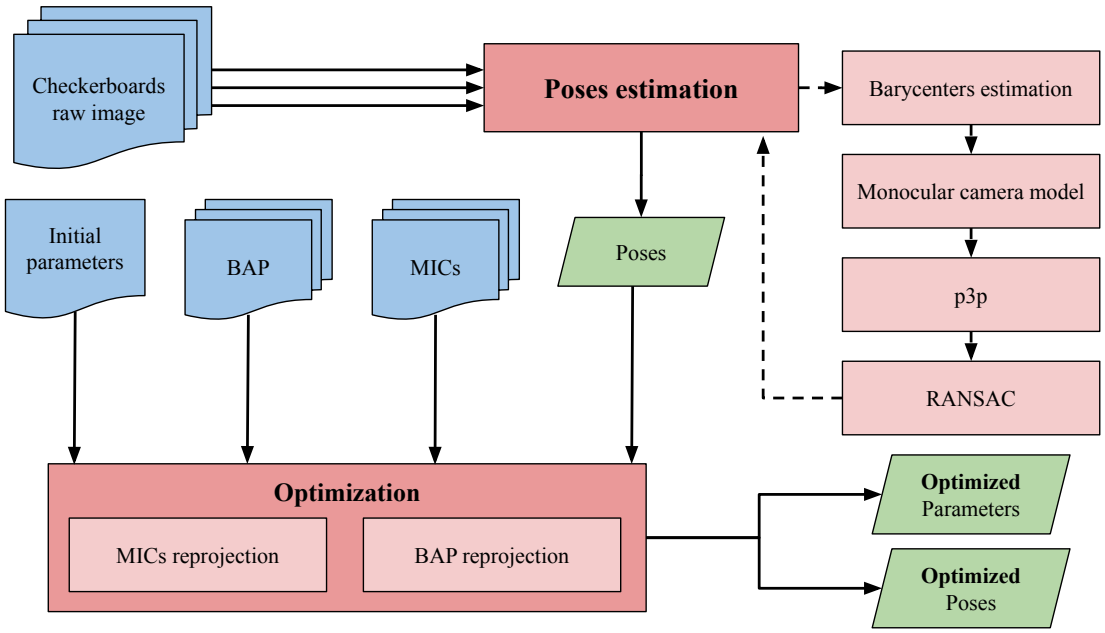


Figure 3.25: Overview of our calibration step. From checkerboard raw images, poses are estimated. With initial parameters and features, our non-linear optimization minimizes the MICs and the BAP reprojection errors to obtain the optimized poses and parameters of our model.

3.3.4 Calibration of plenoptic cameras

To retrieve the intrinsic parameters Ξ of our camera model (see Eq. (3.40)), we use a calibration process based on non-linear minimization of reprojection errors. It is illustrated in Figure 3.25. The camera calibration process is divided into three phases:

1. the initial intrinsics are provided by the pre-calibration step;
2. the initial extrinsics are estimated from the raw checkerboard images;
3. the parameters are refined with a non-linear optimization leveraging our new BAP features.

3.3.4.1 Camera model initialization

Iterative optimization of non-linear cost functions are sensitive to initial parameters installation. To ensure convergence and to avoid falling into local minima during the process, the parameters must be carefully initialized close to the solution. Our pre-calibration step provides a strong initial solution for the optimization. Intrinsic parameters are initialized as explained in subsubsection 3.3.2.3 using only raw white images. The camera poses $\{\boldsymbol{T}_c^n\}$, i.e., the extrinsic parameters, are initialized using

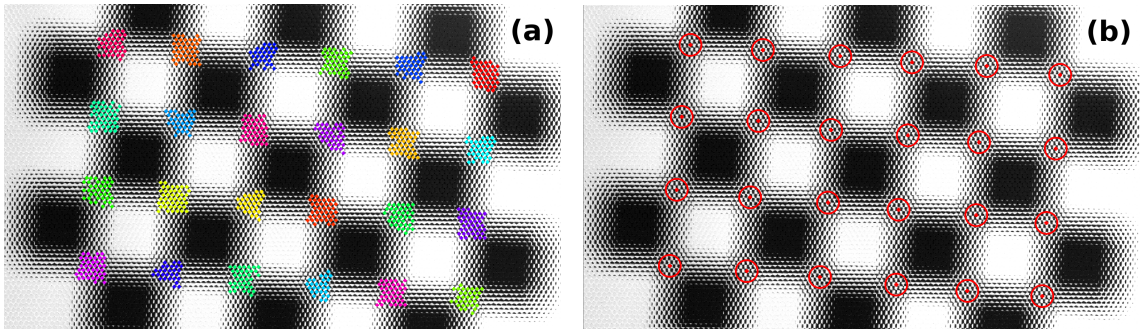


Figure 3.26: Checkerboard raw image with: **(a)** clusters of observations; **(b)** their barycenter used as approximation for extrinsics initialization.

the same method as by Noury *et al.* [166]. We compute the barycenter of each cluster of observations as illustrated by Figure 3.26. Those barycenters can been seen as the projections of the checkerboard corners through the main lens using a standard pinhole model. For each frame, the pose is then estimated using the PnP algorithm [179], like in classic pinhole imaging systems. To associate 3D-2D correspondences, we reproject checkerboard corners based on the estimated pose in image space and link them to their nearest cluster of observations.

3.3.4.2 Optimizing the camera parameters

We propose a new cost function Θ taking into account the blur information of our new BAP feature. The cost is composed of two main terms both expressing errors in the image space:

1. the *blur aware plenoptic reprojection error*,
2. the main lens center reprojection error.

In the first term, for each frame n, each checkerboard corner $\boldsymbol{p}_w^n$ is reprojected into the image space through each micro-lens (k,l) of type (i) according to the projection model of Eq. (3.40) and compared to its observations $\boldsymbol{p}_{k,l}^n$. In the second term, the main lens center $\boldsymbol{O}$ is reprojected according to a pinhole model in the image space through each micro-lens (k,l) and compared to its detected micro-image center $\boldsymbol{c}_{k,l}$. Let $\mathcal{S} = \{\Xi, \{\boldsymbol{T}_c^n\}\}$ be the set of intrinsic Ξ and extrinsic $\{\boldsymbol{T}_c^n\}$ parameters to be optimized. The cost function $\Theta(\mathcal{S})$ is expressed as

$$\Theta(\mathcal{S}) = \sum \left\| \boldsymbol{p}_{k,l}^n - \Pi_{k,l}\left(\boldsymbol{p}_w^n\right) \right\|^2 + \sum \left\| \boldsymbol{c}_{k,l} - \Pi_{k,l}\left(\boldsymbol{O}\right) \right\|^2 . \tag{3.68}$$

The optimization is conducted using the Levenberg-Marquardt algorithm.

3.3.4.3 Inverse distortion calibration

The optimization of the inverse coefficients is done only once, as a post-calibration step, such that

$$\underset{\{Q_{\text{-}1},Q_{\text{-}2},Q_{\text{-}3},P_{\text{-}1},P_{\text{-}2}\}}{\arg\min} \sum_{\boldsymbol{p}} \left\| \boldsymbol{p} - \varphi^{-1}(\varphi(\boldsymbol{p})) \right\|^2, \tag{3.69}$$

for a large number of samples $\boldsymbol{p}$ uniformly distributed in the virtual intermediate space. The optimization is also conducted with the Levenberg-Marquardt algorithm.

3.3.5 Relative blur calibration

In parallel, we leverage the relative blur between different micro-images and use our BAP features to calibrate the blur proportionality coefficient κ of Eq. (3.23). It is done by minimizing the relative blur in a new reprojection error with a non-linear optimization. Relative blur estimation has been studied by Ens and Lawrence [180] and Mannan and Langer [181]. Up to our knowledge, it has never been studied in context of plenoptic camera.

3.3.5.1 Relative blur model

A point imaged by two different micro-lenses of type (i) and (j) will give different blur amounts, i.e., the resulting images will have different spread parameters for the PSF model, such as

$$\begin{cases} \mathcal{I}_{(i)}(x,y) = \mathrm{h}_{(i)} * \mathcal{I}^*(x,y) \\ \mathcal{I}_{(j)}(x,y) = \mathrm{h}_{(j)} * \mathcal{I}^*(x,y)\,, \end{cases} \tag{3.70}$$

where $\mathcal{I}^*(x,y)$ is the latent in-focus image. We approximate the PSF with a 2D Gaussian as in Eq. (3.22), where the diameter of the blur kernel $\mathrm{h}_{(i)}$ is $\sigma_{(i)}$. To compare two views with different amounts of blur, we use the relative blur model in spatial domain [132], [177], [180], [182]. As stated by Mannan and Langer [183], the Gaussian relative blur approximation works well mainly for small relative blurs (up to $\rho \approx 5$ pixels) and when the aperture has a simple shape, which is the case with the plenoptic camera. We then use the equally-defocused representation, in a similar manner as [184], by applying additional blur to the relatively in-focus micro-image,

$$\begin{cases} \mathcal{I}_{(i)}(x,y) \simeq \mathrm{h}_r * \mathcal{I}_{(j)}(x,y) & \text{if } \sigma_{(i)} > \sigma_{(j)} \\ \mathrm{h}_r * \mathcal{I}_{(i)}(x,y) \simeq \mathcal{I}_{(j)}(x,y) & \text{if } \sigma_{(i)} \leq \sigma_{(j)} \end{cases}. \tag{3.71}$$

Note that h_r is the relative blur kernel applied to either one of the views such that both views are equally-defocused. The diameter of the relative blur kernel h_r is approximated as

$$\sigma_r(i,j) \simeq \sqrt{|\sigma_{(i)}^2 - \sigma_{(j)}^2|}. \tag{3.72}$$

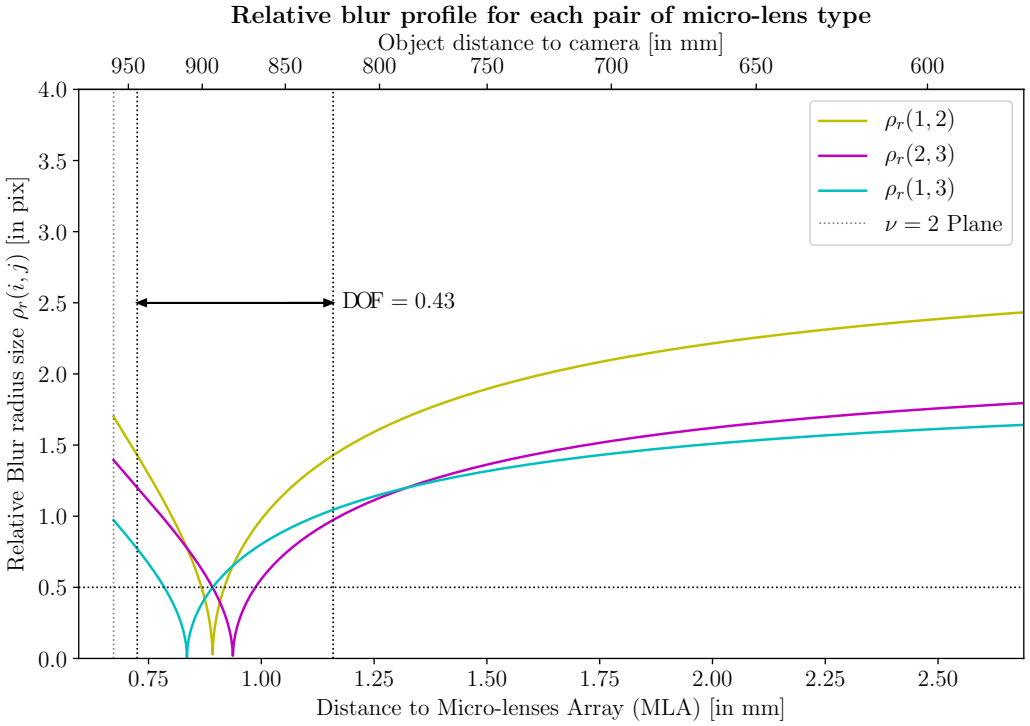


Figure 3.27: Relative blur radius profiles of each pair of micro-lens type, in MLA space, with a focus distance $h = 1000$ mm (from calibration of dataset `R12-B`). The relative blur radius is expressed in pixel as function of the distance to the MLA in mm.

This approximation is exact when the PSF is Gaussian. Since the radius of the relative blur kernel σ_r cannot indicate whether the (i) or the (j) view is more in-focus than the other, we define the relative blur as

$$\Delta\sigma^2(i,j) \triangleq \sigma^2_{(i)} - \sigma^2_{(j)}, \tag{3.73}$$

where $\Delta\sigma^2(i,j) > 0$ indicates that a pixel in the (j)-micro-image is more in-focus than its corresponding pixel in the (i)-micro-image, and vice-versa. Similarly, we define the *relative blur radius* as

$$\rho_r(i,j) \simeq \sqrt{|\Delta\rho^2(i,j)|} = \sqrt{|\rho^2_{(i)} - \rho^2_{(j)}|} \tag{3.74}$$

with $\sigma_r = \kappa \cdot \rho_r$, and where $\rho_{(i)}, \rho_{(j)}$ are the blur radii of the BAP features through a micro-lens of type (i) and (j). The characterization of relative blur profile for each pair of micro-lens type $((1,2),(1,3),(2,3))$ of the plenoptic camera is illustrated by the Figure 3.27. It corresponds to a configuration at focus distance $h = 1000$ mm (dataset `R12-B`).

3.3.5.2 Blur proportionality coefficient calibration

To calibrate the blur proportionality coefficient κ, we use our BAP features and the relative blur model applied on micro-images of different types. The BAP features

$\{\boldsymbol{p}_i\}$ from a same cluster $\mathcal{C}$ represent the same point in object space $\boldsymbol{p}_w$. We extract two windows $\mathcal{W}$ around the BAP features $\boldsymbol{p}_i, \boldsymbol{p}_j \in \mathcal{C}(\boldsymbol{p}_w)$ of different types, and express them using the equally-defocused representation (Eq. (3.71)). The relative blur radius does not exceed 2.5 pixel. So windows $\mathcal{W}$ of size 9×9 are large enough to capture all the information. They are extracted at (u, v) with sub-pixel precision, and represent therefore the same part of the scene in both micro-images. Additional blur is applied using a Gaussian kernel of spread parameter σ_r. The spread parameter is computed from the ρ part of the BAP features and the parameter κ to be optimized, with initial value $\kappa = 1$. Let $\Theta(\kappa)$ be the cost function to be minimized. It is expressed as

$$\Theta(\kappa) = \sum_{n} \sum_{\boldsymbol{p}_i^n, \boldsymbol{p}_j^n \in \mathcal{C}(\boldsymbol{p}_w^n)} \left\| \mathcal{W}(\boldsymbol{p}_j^n) - \mathrm{h}_r * \mathcal{W}(\boldsymbol{p}_i^n) \right\|_2^2, \tag{3.75}$$

given $\left|\rho_{(i)}\right| < \left|\rho_{(j)}\right|$ and where h_r is the PSF with spread parameter

$$\sigma_r = \kappa \cdot \sqrt{|\rho_{(i)}^2 - \rho_{(j)}^2|}.$$

The optimization is conducted using the Levenberg-Marquardt algorithm.

3.4 Experimental validation

In this section, we will present the experimental validation of our proposed camera model and calibration method. First, we will detail the experimental setup, based on real-data obtained in a controlled environment as well as in simulation, for several camera setups. Second, we will give and discuss the results regarding the pre-calibration and the evaluations of the calibration parameters obtained by our method and compared to state-of-the-art methods. A focus on the relative blur calibration will be done, and we will conclude by an ablation study of the proposed model.

3.4.1 Experimental setup

To validate our camera model, we evaluate our method on real-world data obtained with a multi-focus plenoptic camera in a controlled environment. Our experimental setup is illustrated in Figure 3.28. The camera is mounted on a linear motion table with micro-metric precision. The target plane is orthogonal to the translation axis, and the camera optical axis is aligned with this axis. The approximate absolute distances at which the images have been taken with the corresponding step lengths are reported in Table 3.3.

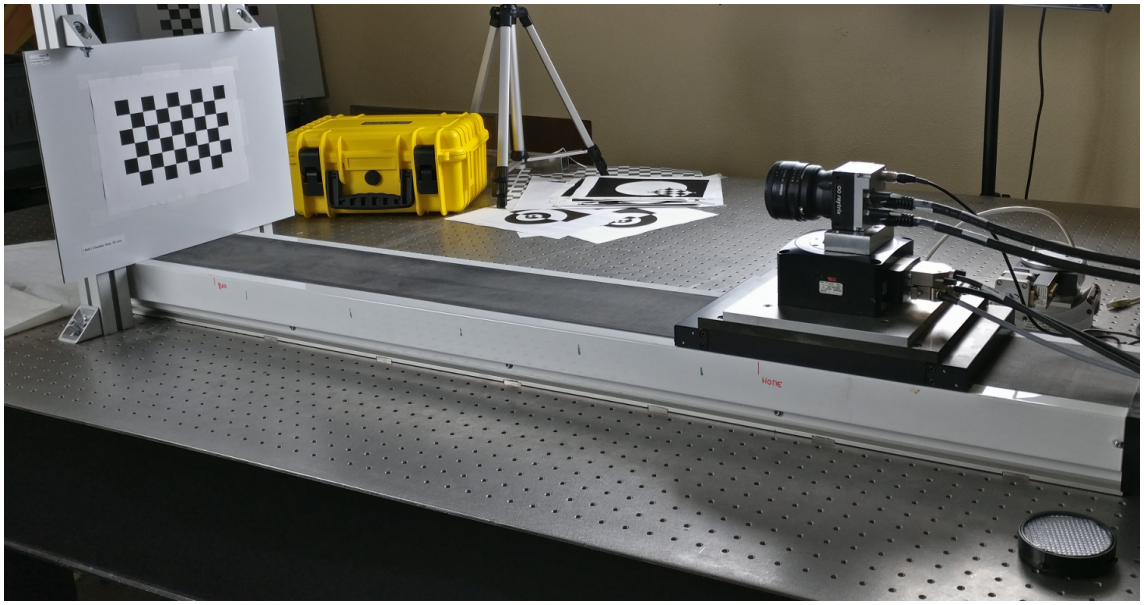


Figure 3.28: The `Raytrix R12` multi-focus plenoptic camera used in our experimental setup. The camera is mounted on a linear motion table with micro-metric precision.

3.4.1.1 Hardware environment

For our experiments we used a `Raytrix R12` color 3D-light-field-camera, with a MLA of F/2.4 aperture. The camera is in Galilean *internal* configuration. We used two different mounted lenses, a `Nikon AF Nikkor F/1.8D` with a 50 mm focal length for comparison with state-of-the-art, and a `Nikon AF DC-Nikkor F/2D` with a 135 mm focal length to validate the generalization of our model. The MLA organization is hexagonal row-aligned, and composed of 176×152 (width $\times$ height) micro-lenses with $I = 3$ different types. The sensor is a `Basler beA4000-62KC` with a pixel size of $s = 0.0055$ mm. The raw image resolution is 4080×3068 pixel. We calibrate our camera for four focus distance configurations, with $h \in \{450, 1000, \infty\}$ mm for the 50 mm lens, and with $h = 1500$ mm for the 135 mm lens. Note that when changing the focus setting, the main lens moves with respect to the block MLA-sensor.

3.4.1.2 Software environment

All images have been acquired using the `MultiCamStudio` free software (v6.15.1.3573) of the `Euresys` company. We set the shutter speed to 5 ms. While taking white images for the pre-calibration step, we set the gain to its maximum value. For `Raytrix` data, we used their proprietary software `RxLive` (v4.0.50.2) to calibrate the camera, and computed the depth maps used in the evaluation. Our source code has been made publicly available: https://github.com/comsee-research/libpleno, and https://github.com/comsee-research/compote.

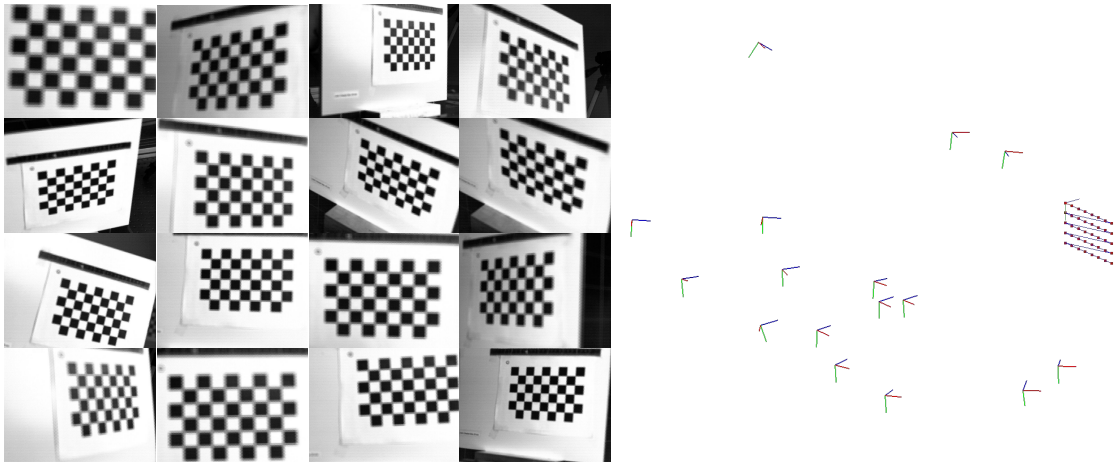


Figure 3.29: Example of calibration targets acquired for distances between 775 and 400 mm from the checkerboard used in the dataset `R12-B`, and their respective poses in 3D.

3.4.1.3 Datasets

We build five real-world datasets with different focus distance h: for the 50 mm lens, `R12-A` for $h = 450$ mm, `R12-B` for $h = 1000$ mm, `R12-C` for $h = \infty$, and `R12-E` for $h = 2133$ mm; for the 135 mm lens, `R12-D` for $h = 1500$ mm. Each dataset is composed of:

- white raw plenoptic images acquired at different apertures ($N \in \{4, 5.66, 8, 11.31, 16\}$) using a light diffuser mounted on the main objective for pre-calibration,
- free-hand calibration target images acquired at various poses (in distance and orientation), separated into two subsets, one for the calibration process (16 images) and the other for reprojection error evaluation (15 images),
- a white raw plenoptic image acquired in the same luminosity condition and with the same aperture as in the calibration targets acquisition for devignetting,
- calibration targets acquired with a controlled translation motion for quantitative evaluation, along with the depth maps computed by `RxLive`.

The dataset `R12-E` contains only free-hand calibration target real images. Evaluation targets are generated in simulation. More details about this dataset will be provided in section 4.4.1. Examples of calibration targets acquired for the `R12-B` dataset are given in Figure 3.29 along with their 3D poses. A summary for each dataset is given in Table 3.3, indicating checkerboard information and the distances at which the targets have been acquired for calibration and for the controlled evaluation. Our datasets have been made publicly available, and can be downloaded from our public repository at https://github.com/comsee-research/plenoptic-datasets. For more details, see Appendix B.

Table 3.3: Summary of `R12-A,B,C,D,E`, and `UPC-S` datasets contents. All distances are given in mm. Scale refers to checkerboard square size. Evaluation distances refer to the linear motion table setup.

		Target information		Calibration distances		Evaluation distances		
	h	size	scale	min	max	min	max	step
`A`	450	9×5	10	175	400	265	385	10
`B`	1000	8×5	20	400	775	450	900	50
`C`	∞	6×4	30	500	2500	400	1250	50
`D`	1500	5×3	20	850	1300	750	1200	50
`S`	hyperf.	9×6	26.25	250	800	200	500	50
`E`	2133	6×4	30	350	1600	500	1900	100

3.4.1.4 Simulation environment

In order to validate our model on *unfocused* plenoptic camera (UPC), i.e., `Lytro`-like plenoptic camera configuration, we propose to evaluate our model in a simulation environment. We built our own simulator based on raytracing to generate images, named `prism` (see Appendix C). Similar to the real-world dataset, we generated a dataset, named `UPC-S`, composed of several white images taken at different apertures (with $N \in \{2, 4, 5.6\}$), various checkerboard poses for calibration and validation, and for evaluation, checkerboard images with known translation along the z-axis. An example of a generated image is given in Figure 3.30. Details are also given in Table 3.3. We used the `Lytro Illum` intrinsic parameters reported in [149, Table 4] as baseline for the simulation. They have been converted into our parameters and reported in Table 3.6. The MLA arrangement is hexagonal row-aligned, and composed of 541×434 (width $\times$ height) micro-lenses of the same type ($I = 1$). The raw image resolution is 7728×5368 pixel, with a pixel size of $s = 0.0014$ mm and with micro-image of radius 7.172 pixel.

3.4.2 Calibrations results

Our evaluation process follows the steps given in the overview (Figure 3.13). First, we present the pre-calibration results, where white raw plenoptic images are used for computing micro-image centers, and for estimating initial camera parameters. Second, from the set of devignetted calibration target images, BAP features are extracted, and camera intrinsic and extrinsic parameters are then estimated using our non-linear optimization process. In parallel, the same BAP features are also used to calibrate the relative blur proportionality coefficient. Finally, we evaluate

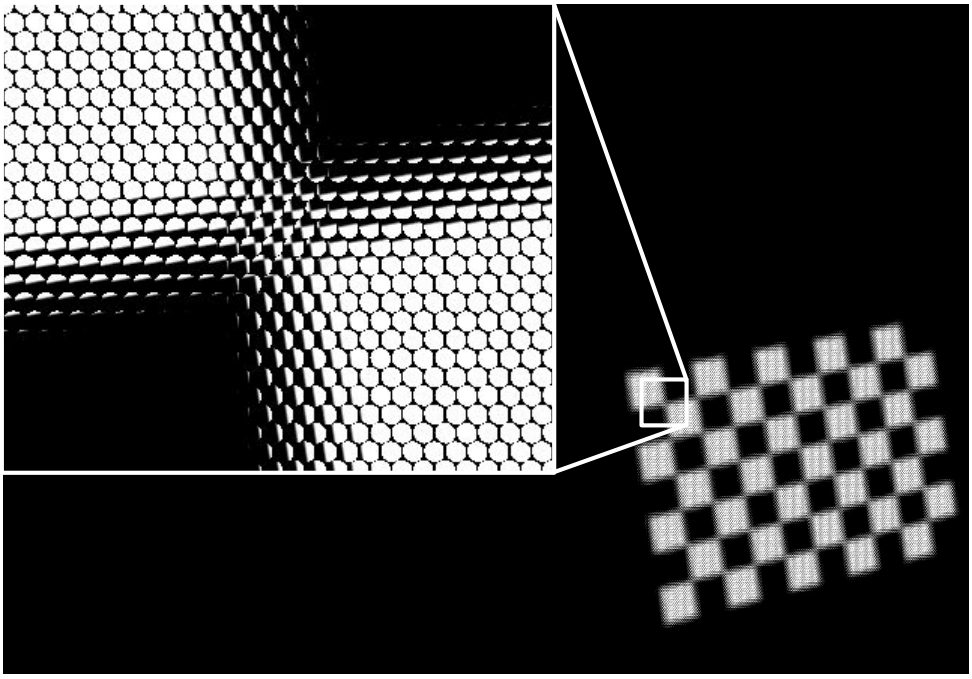


Figure 3.30: Example of raw image of a checkerboard generated by our simulator `prism`, part of our dataset `UPC-S`.

our model quantitatively, first, using the reprojection error as a metric, and second, using the relative translation error in a controlled environment.

3.4.2.1 Pre-calibration

To estimate the parameters Ω, we set $\alpha = 2.357$, and since the camera is in Galilean *internal* configuration, we use $R = -\varrho \cdot s$, following Eq. (3.55). Figure 3.21 shows the micro-image radii as function of the inverse f-number with the estimated lines for dataset `R12-B`. Their distributions are represented by the violin-boxes. For $N = 5.66$, we can see that radii distributions overlap, and that radii values are slightly overestimated as they do not fit exactly the borders of the micro-images (see Figure 3.20). In practice, we only use white images that present distinguishable radii distributions in the estimation process, usually corresponding to small apertures. For instance, in case of `R12-B`, only white images at $N = 11.31$ and $N = 8$ are used. The corresponding coefficients for all datasets are summarized in Table 3.4. As expected, the parameter m is different for each dataset, since D and Δ_i vary with the focus distance h, whereas the q'_i values are close for all datasets, even for different camera setup (`R12-D`). For the specific case of the *unfocused* plenoptic camera, i.e., `UPC-S`, the pre-calibration step has also be done, with $I = 1$. We expect the initial value of f to be equals (or at least approximately equals) to the distance d. From Eq. (3.50), we should have $q' = \Delta_i/2$. The reported value is $q' = 9.894 \approx \Delta_i/2 = 9.993$ µm which conforms to the hypothesis $f = d$.

Table 3.4: Set of parameters Ω (in µm) computed during the pre-calibration step for each dataset, along with the calibrated relative blur proportionality coefficient.

	R12-A	R12-B	R12-C	R12-D	R12-E	UPC-S
Δ_i	128.225	128.288	128.326	127.851	128.3211	20.0814
λ	0.99407	0.99370	0.99352	0.99746	0.99348	0.99529
m	−149.202	−158.596	−163.136	−171.288	−164.00	−23.6396
q'_1	37.221	37.201	36.902	38.599	36.562	9.8943
q'_2	41.404	41.569	41.575	43.129	39.139	-
q'_3	38.695	38.844	38.771	40.788	37.694	-
κ	0.8134	0.7763	0.7404	0.8824	1.0202	-

Table 3.5: Statistics (mean±std) over radii measurements (in µm) for each type (i) of micro-image at different apertures for the dataset R12-A.

Type	$N = 5.66$	$N = 8$	$N = 11.31$
$i = 1$	56.79±0.56	47.08±0.53	42.76±1.04
$i = 2$	53.68±0.95	41.71±0.78	35.68±1.36
$i = 3$	56.58±0.91	44.46±0.73	38.98±1.32

In addition, an analysis of the micro-image radii distribution is given for three apertures $N \in \{5.66, 8, 11.31\}$ in Table 3.5 for the dataset R12-A. As expected, the radius decreases whilst the f-number N increases. The standard deviation is less than one-fifth of a pixel, meaning that our method provides precise results. However, we can note that more the aperture decreases, the less the distributions are distinguishable from each others.

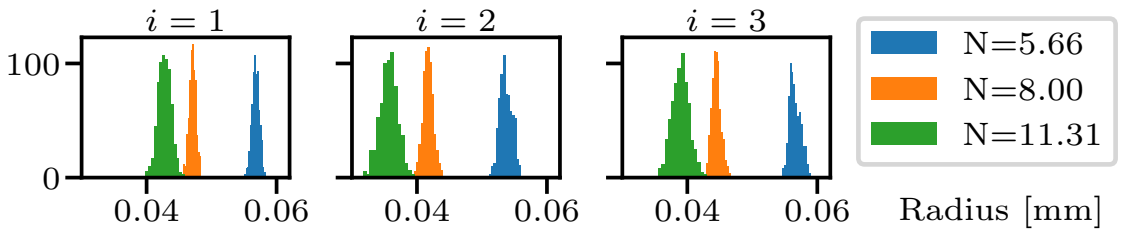


Figure 3.31: Distribution of radii measurements for each type (i) of micro-image at different apertures for the dataset R12-A.

3.4.2.2 Free-hand camera calibration

Comparison with state-of-the-art. Since our model is close to the one of Noury *et al.* [166], we compare our intrinsics with the ones obtained under their pinholes assumption using only corner reprojection error and with the same initial parameters. In addition, we evaluate against the method of Nousias *et al.* [161], which provides a set of intrinsics and extrinsics for each micro-lens type. The equivalence of our parameters and their parameters is given by

$$\begin{cases} F = \dfrac{(f_x + f_y)}{2} \cdot s, \quad D = -F \cdot \left(\dfrac{K_1}{K_2} \cdot F + 1 \right)^{-1}, \\ d^{(i)} = D - \dfrac{K_2 D}{D + K_2}, \quad u_0 = c_x \quad \text{and} \quad v_0 = c_y, \end{cases} \tag{3.76}$$

where K_1 and K_2 are the two additional intrinsic parameters that account for the MLA setting in their model. The equivalence also stands for the parameters of Bok *et al.* [149]. The provided detector from Nousias *et al.* [161] was not able to detect corner observations on our datasets. Therefore, we used the same observations for our method (noted `BAP`), Noury *et al.* [166] method (`NOUR`), and Nousias *et al.* [161] method for each type (`NOUS1`, `NOUS2`, and `NOUS3`), which allowed us to focus the comparison on the camera model only. Finally, we provide the calibration parameters obtained from the `RxLive` software (`RTRX`) corresponding to the model of Heinze *et al.* [125], and compare our depth estimates to their depth maps.

Initialization. We initialize λ from Eq. (3.58). Its value for each dataset is reported in Table 3.4. The difference between the initial value of λ and its value computed from optimized camera parameter is less than 0.024 %, which validates the use of the initial value from Eq. (3.58) when computing our BAP features. The initial camera parameters reported in Tables 3.7, 3.8, 3.9, and 3.10, are computed using the methodology presented in subsubsection 3.3.2.3. They are used for the `BAP` and `NOUR` methods. The camera *internal* configuration is set to Galilean. When h decreases, D increases. Yet when the main lens focus distance is at infinity, the main lens should focus on the plane $\upsilon = 2$, which implies that D tends to $F - 2d$ as lower bound, as H tends to F. In most cases (here, for `R12-A,B,D,E`), we will still have $F < D$, which usually can describe the camera in Keplerian configuration. In Keplerian *internal* configuration, the condition $F < D$ stands regardless of the focus distance, as D lower bound is $F + 2d$.

When using the linear initialization from `NOUS`, the initial parameters of some configurations corresponded to impossible physical setup or were too far from the solution, hindering the convergence of the optimization. Therefore, in order to continue comparison, we manually set the initial parameters close enough to a solution. In contrast, we can see that the optimized parameters for `BAP` and `NOUR` are close to initial values, which shows that our pre-calibration step provides a strong initial solution for the optimization process.

Intrinsic camera parameters. Optimized intrinsic parameters are reported for each dataset and for all the evaluated methods in Tables 3.7, 3.8, 3.9, and 3.10. First, `BAP`, `NOUR` and `NOUS` all verify the condition $F \approx D + 2d$ when the focus is set at infinity (`R12-C`). Second, the focal lengths obtained from `NOUR`, `NOUS` and `RTRX` change significantly given the focus distance, and the ones obtained from `NOUS` even vary according to the micro-lens types. In contrast, only `BAP` shows stable parameters across all four `R12-A,B,C,E` datasets. Shared parameters across datasets (i.e., the focal lengths and the distance between the MLA and the sensor) are close enough to indicate that our model successfully generalizes to different focus configurations. Furthermore, the parameters obtained by our method with an other main lens, i.e., `R12-D`, are coherent with the previously obtained parameters, stressing out that our model can be applied to a different camera setting. Finally, our method is the only one providing the micro-lenses focal lengths in a single unified model. The other methods calibrate either several MLA-sensor distances (`RTRX`), or several models, one for each type (`NOUS`). Note that distortion coefficients and MLA rotations are close to zero. The influence of these parameters will be analyzed in the proposed ablation study of the camera model in section 3.4.3.

On simulated data. First, pre-calibration has been performed using the white raw images. The resulting parameters Ω are coherent with the simulation parameters. Reference and initial intrinsic parameters are reported in Table 3.6, along with the optimized parameters. Second, calibration has been performed. The obtained intrinsic parameters are close enough to the references parameters, indicating that our method is able to generalize to the *unfocused* plenoptic camera. For completeness, we also quantitatively evaluated the optimized parameters, by estimating the relative displacement between checkerboards with known motion along the z-axis. It results a translation error $\varepsilon_z = 1.64\,\%$, which validates the model.

Table 3.6: Reference and initial intrinsic parameters for the simulated `Lytro` dataset `UPC-S` along with the optimized parameters obtained by our method (`BAP`).

		Reference	Initial	`BAP`
F	[mm]	9.9845	10	10.230
Δ_μ	[µm]	20	19.987	19.987
D	[mm]	9.8479	10	10.005
d	[µm]	40.087	47.279	47.323
f	[µm]	40.087	47.753	47.747
u_0	[pix]	3842.8	3863	3861.7
v_0	[pix]	2719.5	2683	2715.5

Table 3.7: Initial intrinsic parameters for dataset `R12-A` along with the optimized parameters obtained by our method (`BAP`) and with the methods of Noury *et al.* [166] (`NOUR`), of Nousias *et al.* [161] for each micro-lens type (`NOUS1, NOUS2, NOUS3`) and the parameters obtained from `RxLive` software (`RTRX`).

		`R12-A` ($h = 450$ mm)						
		Init.	`BAP`	`NOUR`	`NOUS1`	`NOUS2`	`NOUS3`	`RTRX`
F	[mm]	50	49.885	54.761	61.305	62.476	63.328	47.709
Q_1	[$\times 10^{-5}$]	0	24.63	6.194	-	-	-	-
$-Q_2$	[$\times 10^{-6}$]	0	3.032	0.800	-	-	-	-
Q_3	[$\times 10^{-8}$]	0	1.095	0.252	-	-	-	-
P_1	[$\times 10^{-5}$]	0	-11.1	-18.1	-	-	-	-
$-P_2$	[$\times 10^{-5}$]	0	3.599	5.186	-	-	-	-
$-Q_{\text{-}1}$	[$\times 10^{-5}$]	0	24.29	-	-	-	-	-
$Q_{\text{-}2}$	[$\times 10^{-6}$]	0	2.971	-	-	-	-	-
$-Q_{\text{-}3}$	[$\times 10^{-8}$]	0	1.066	-	-	-	-	-
$-P_{\text{-}1}$	[$\times 10^{-5}$]	0	-10.89	-	-	-	-	-
$P_{\text{-}2}$	[$\times 10^{-5}$]	0	3.540	-	-	-	-	-
D	[mm]	56.619	56.860	62.341	71.131	72.541	73.530	-
$-t_x$	[mm]	11.29	10.93	9.480	-	-	-	-
$-t_y$	[mm]	8.410	7.996	8.087	-	-	-	-
$-\theta_x$	[µrad]	0	388.9	460.3	-	-	-	-
θ_y	[µrad]	0	271.4	363.4	-	-	-	-
θ_z	[µrad]	14.9	29.5	25.6	-	-	-	41.9
Δ_μ	[µm]	127.46	127.46	127.40	-	-	-	127.36
$f^{(1)}$	[µm]	578.58	582.67	-	-	-	-	-
$f^{(2)}$	[µm]	520.14	524.02	-	-	-	-	-
$f^{(3)}$	[µm]	556.54	560.57	-	-	-	-	-
u_0	[pix]	2039	2078.3	2343.4	1984.9	2034.5	1973.7	-
v_0	[pix]	1533	1591.0	1573.7	1482.1	1481.0	1495.2	-
d	[µm]	337.91	337.13	391.90	-	-	-	-
$d^{(1)}$	[µm]	-	-	-	585.16	-	-	407.81
$d^{(2)}$	[µm]	-	-	-	-	527.59	-	406.00
$d^{(3)}$	[µm]	-	-	-	-	-	561.93	406.90

Table 3.8: Initial intrinsic parameters for dataset `R12-B` along with the optimized parameters obtained by our method (`BAP`) and with the methods of Noury *et al.* [166] (`NOUR`), of Nousias *et al.* [161] for each micro-lens type (`NOUS1, NOUS2, NOUS3`) and the parameters obtained from `RxLive` software (`RTRX`).

		`R12-B` ($h = 1000$ mm)						
		Init.	`BAP`	`NOUR`	`NOUS1`	`NOUS2`	`NOUS3`	`RTRX`
F	[mm]	50	50.011	51.177	53.913	52.988	52.977	50.894
Q_1	[$\times 10^{-5}$]	0	4.661	1.650	-	-	-	-
$-Q_2$	[$\times 10^{-6}$]	0	0.516	0.264	-	-	-	-
Q_3	[$\times 10^{-8}$]	0	0.156	0.078	-	-	-	-
P_1	[$\times 10^{-5}$]	0	12.84	11.27	-	-	-	-
$-P_2$	[$\times 10^{-5}$]	0	24.33	23.16	-	-	-	-
$-Q_{\text{-}1}$	[$\times 10^{-5}$]	0	4.685	-	-	-	-	-
$Q_{\text{-}2}$	[$\times 10^{-6}$]	0	0.528	-	-	-	-	-
$-Q_{\text{-}3}$	[$\times 10^{-8}$]	0	0.160	-	-	-	-	-
$-P_{\text{-}1}$	[$\times 10^{-5}$]	0	12.86	-	-	-	-	-
$P_{\text{-}2}$	[$\times 10^{-5}$]	0	24.47	-	-	-	-	-
D	[mm]	52.125	52.140	53.213	56.062	55.128	55.124	-
$-t_x$	[mm]	11.30	12.15	12.38	-	-	-	-
$-t_y$	[mm]	8.415	6.165	5.965	-	-	-	-
$-\theta_x$	[µrad]	0	488.4	555.4	-	-	-	-
θ_y	[µrad]	0	286.5	330.1	-	-	-	-
θ_z	[µrad]	14.7	30.9	33.9	-	-	-	41.9
Δ_μ	[µm]	127.48	127.47	127.41	-	-	-	127.36
$f^{(1)}$	[µm]	566.57	566.39	-	-	-	-	-
$f^{(2)}$	[µm]	507.03	507.09	-	-	-	-	-
$f^{(3)}$	[µm]	542.61	542.47	-	-	-	-	-
u_0	[pix]	2039	1855.8	1811.9	2074.7	2094.7	1837.0	-
v_0	[pix]	1533	1926.2	1962.2	1640.2	1649.1	1620.4	-
d	[µm]	330.67	326.72	361.01	-	-	-	-
$d^{(1)}$	[µm]	-	-	-	447.81	-	-	407.81
$d^{(2)}$	[µm]	-	-	-	-	401.93	-	406.00
$d^{(3)}$	[µm]	-	-	-	-	-	414.32	406.90

Table 3.9: Initial intrinsic parameters for dataset `R12-C` along with the optimized parameters obtained by our method (`BAP`) and with the methods of Noury *et al.* [166] (`NOUR`), of Nousias *et al.* [161] for each micro-lens type (`NOUS1, NOUS2, NOUS3`) and the parameters obtained from `RxLive` software (`RTRX`).

		R12-C $(h = \infty)$						
		Init.	BAP	NOUR	NOUS1	NOUS2	NOUS3	RTRX
F	[mm]	50	50.099	51.644	51.113	49.919	50.812	51.564
Q_1	[$\times 10^{-5}$]	0	13.84	1.292	-	-	-	-
$-Q_2$	[$\times 10^{-6}$]	0	2.723	0.576	-	-	-	-
Q_3	[$\times 10^{-8}$]	0	1.260	0.185	-	-	-	-
P_1	[$\times 10^{-5}$]	0	2.51	12.13	-	-	-	-
$-P_2$	[$\times 10^{-5}$]	0	-3.072	-0.027	-	-	-	-
$-Q_{\text{-}1}$	[$\times 10^{-5}$]	0	13.78	-	-	-	-	-
$Q_{\text{-}2}$	[$\times 10^{-6}$]	0	2.705	-	-	-	-	-
$-Q_{\text{-}3}$	[$\times 10^{-8}$]	0	1.246	-	-	-	-	-
$-P_{\text{-}1}$	[$\times 10^{-5}$]	0	2.50	-	-	-	-	-
$P_{\text{-}2}$	[$\times 10^{-5}$]	0	-3.067	-	-	-	-	-
D	[mm]	49.356	49.356	50.728	50.331	49.067	49.882	-
$-t_x$	[mm]	11.30	12.53	13.24	-	-	-	-
$-t_y$	[mm]	8.417	8.237	7.400	-	-	-	-
$-\theta_x$	[µrad]	0	409.8	442.2	-	-	-	-
θ_y	[µrad]	0	306.1	333.4	-	-	-	-
θ_z	[µrad]	37.2	33.9	39.9	-	-	-	36.6
Δ_μ	[µm]	127.49	127.46	127.41	-	-	-	127.36
$f^{(1)}$	[µm]	556.37	580.80	-	-	-	-	-
$f^{(2)}$	[µm]	493.83	515.57	-	-	-	-	-
$f^{(3)}$	[µm]	529.54	552.84	-	-	-	-	-
u_0	[pix]	2039	1786.6	1654.9	1966.3	1913.8	2052.5	-
v_0	[pix]	1533	1547.1	1699.7	1484.6	1487.2	1492.7	-
d	[µm]	322.07	330.32	357.82	-	-	-	-
$d^{(1)}$	[µm]	-	-	-	357.80	-	-	407.81
$d^{(2)}$	[µm]	-	-	-	-	349.99	-	406.00
$d^{(3)}$	[µm]	-	-	-	-	-	353.26	406.90

Table 3.10: Initial intrinsic parameters for datasets R12-D and R12-E along with the optimized parameters obtained by our method (BAP).

		R12-D ($h = 1500$ mm)		R12-E ($h = 2133$ mm)	
		Init.	BAP	Init.	BAP
F	[mm]	135	136.105	50	50.119
Q_1	[$\times 10^{-5}$]	0	35.974	0	-6.823
$-Q_2$	[$\times 10^{-6}$]	0	8.083	0	-0.408
Q_3	[$\times 10^{-8}$]	0	4.821	0	-0.047
P_1	[$\times 10^{-5}$]	0	-4.31	0	20.749
$-P_2$	[$\times 10^{-5}$]	0	-3.76	0	11.128
$-Q_{\text{-}1}$	[$\times 10^{-5}$]	-	-	0	-6.853
$Q_{\text{-}2}$	[$\times 10^{-6}$]	-	-	0	-0.394
$-Q_{\text{-}3}$	[$\times 10^{-8}$]	-	-	0	-0.037
$-P_{\text{-}1}$	[$\times 10^{-5}$]	-	-	0	21.031
$P_{\text{-}2}$	[$\times 10^{-5}$]	-	-	0	11.325
D	[mm]	149.24	149.10	50.654	50.585
$-t_x$	[mm]	11.30	11.21	11.302	12.876
$-t_y$	[mm]	8.387	8.351	8.417	6.616
$-\theta_x$	[µrad]	0	371.1	0	441.6
θ_y	[µrad]	0	287.0	0	289.2
θ_z	[µrad]	5.8	35.4	32.2	37.6
Δ_μ	[µm]	127.53	127.51	127.60	127.45
$f^{(1)}$	[µm]	625.63	636.06	594.57	601.58
$f^{(2)}$	[µm]	559.91	572.52	536.80	562.19
$f^{(3)}$	[µm]	592.05	604.23	568.41	583.54
u_0	[pix]	2039	2028.7	2039	1722.5
v_0	[pix]	1533	1526.7	1533	1843.6
d	[µm]	378.72	382.30	288.16	340.87

3.4.2.3 Quantitative evaluations of the camera model

Reprojection error. In the absence of ground truth, we first evaluate the intrinsic parameters by estimating the reprojection error using the previously computed intrinsics. We consider only free-hand calibration target images which are not used in the calibration process. We use the root-mean-square error (RMSE) as a metric to evaluate the reprojection error on the corner part of the features, for each dataset. For the BAP method, the corner reprojection part is reported in Table 3.11, as well as the radius reprojection part within parentheses. Regarding the NOUS methods, the original error is expressed using the mean reprojection error (MRE). We converted the final error to the RMSE metric for comparison. Note that the latter method operates separately on each type of micro-lens, meaning that the number of features is not the same as with NOUR and BAP. First, the reprojection error is less than 1 pixel for all methods, for each dataset, demonstrating that the computed intrinsics lead to an accurate reprojection model and can be generalized to images which are not from the calibration set. Second, even though the NOUS method provides the lowest RMSE, it shows a significant discrepancy according to the considered type. The error obtained by our method is sightly higher than the error from NOUR, but this is explained by the fact that our optimization does not aim at minimizing only the corner reprojection error but the radius reprojection error as well. Note that the positional error $\varepsilon_{u,v}$ predominates in the total cost by two orders of magnitude compared to the blur radius error ε_{ρ}, but the latter still helps to constrain our model as shown by the relatively close intrinsics across the datasets.

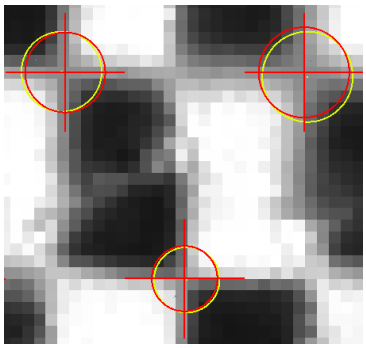


Figure 3.32: Illustration of the reprojection of our BAP features in raw images: in *yellow*, the observations, and in *red*, the reprojections by our model.

Table 3.11: Corner reprojection error for each evaluation dataset (i.e., free-hand calibration target images not part of the calibration dataset) using the RMSE metric. For the BAP method, reprojection error of the radius part is indicated within parentheses.

	BAP	NOUR	NOUS1	NOUS2	NOUS3
R12-A	0.856 (0.083)	0.713	0.773	**0.667**	0.958
R12-B	0.674 (0.183)	0.618	0.538	**0.519**	0.593
R12-C	0.738 (0.041)	0.713	1.287	0.681	**0.411**

Controlled environment poses evaluation. With our experimental setup, we acquired several images with known relative translation between each frame. We

compare the estimated displacements along the z-axis from the extrinsic parameters to the ground truth. The extrinsics are computed with the models estimated from the free-hand calibration. In the case of the `RTRX` method, we use the filtered depth maps obtained with the proprietary software `RxLive` to estimate the displacements. The translation errors along the z-axis with respect to the ground truth displacement from the closest frame are reported in Figure 3.33 for datasets `R12-A` (a), `R12-B` (b) and `R12-C` (c). The relative error ε_z for a known displacement δ_z is computed as the mean absolute relative difference between the estimated displacement $\hat{\delta}_z$ and the ground truth, for each pair of frames $(\boldsymbol{T}_i, \boldsymbol{T}_j)$ separated by a distance δ_z, i.e.,

$$\varepsilon_z(\delta_z) = \eta^{-1} \sum_{(\boldsymbol{T}_i, \boldsymbol{T}_j) | z_i - z_j = \delta_z} \frac{|\delta_z - \hat{\delta}_z|}{\delta_z}, \tag{3.77}$$

where $\hat{\delta}_z = \hat{z}_i - \hat{z}_j$, and η is a normalization constant corresponding to the number of frames pair. The mean error with its standard deviation across all datasets for `BAP`, `NOUR`, `NOUS`, and `RTRX` are reported in (d).

Firstly, the mean error across `R12-A,B,C` datasets are of the same order for the evaluated methods around 3 %:

- for `BAP`, $\varepsilon_z = 2.92 \pm 0.73$ %;
- for `NOUR`, $\varepsilon_z = 3.50 \pm 3.08$ %;
- for `NOUS1`, $\varepsilon_z = 1.68 \pm 1.53$ %;
- for `NOUS2`, $\varepsilon_z = 3.40 \pm 2.19$ %;
- for `NOUS3`, $\varepsilon_z = 3.30 \pm 3.35$ %;
- for `RTRX`, $\varepsilon_z = 4.96 \pm 4.44$ %.

This is also the case for the dataset `R12-D` where our model has a mean translation error of $\varepsilon_z = 3.37$ %. Note that all evaluated methods outperform `RTRX` as the depth maps computation might not be as precise as the optimization of extrinsic parameters. Our method ranks second in terms of relative mean error. Even though lowest error is obtained by the method `NOUS` for type (1), it presents a large standard deviation and the errors for the other two types are significantly higher. In real application context, there is no way to know in advance which type will produce the smallest error. Nousias *et al.* [161] suggested that when extrinsics are sufficiently close, we can use representative extrinsics that are calculated by averaging the extrinsics from the individual types. Our results do not match this observation as the estimated extrinsics are significantly different for each type. As shown, only the first type gives satisfactory results whereas the other two present larger errors with significant standard deviations. Averaging the extrinsics from all types will therefore minimize the difference between poses but will not provide the best possible estimation.

Secondly, the standard deviation can be seen as an indicator of the estimation precision across the datasets, and thus indicates whether the model can generalize to several configurations or not. Our model presents the lowest standard deviation as

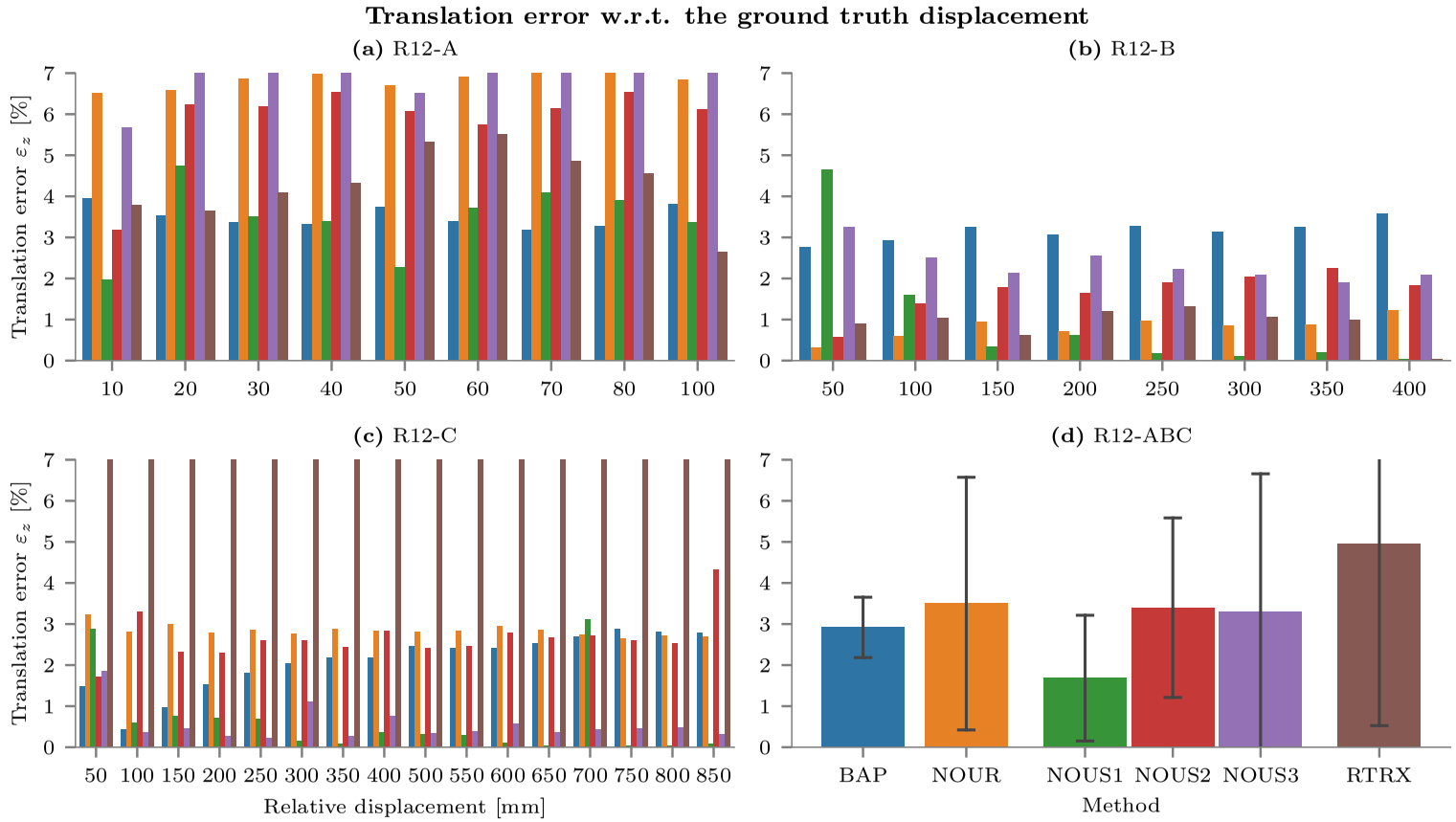


Figure 3.33: Translation error along the z-axis with respect to the ground truth displacement from the closest frame, for datasets `R12-A` **(a)**, `R12-B` **(b)** and `R12-C` **(c)**. The error ε_z is expressed in percentage of the estimated distances, and truncated to 7 % to ease the readability and the comparison. The mean error with its confidence interval across all datasets for our method (`BAP`), Noury *et al.* [166] method (`NOUR`), Nousias *et al.* [161] method for each type (`NOUS1, NOUS2, NOUS3`), and for the proprietary software `RxLive` (`RTRX`) are reported in **(d)**. Please refer to the color version for better visualization.

illustrated in Figure 3.33 (d). This indicates a low discrepancy between datasets and thus that the model is precise and consistent for all configurations.

Thirdly, we analyze the behavior of each method for each dataset across different distances. None of the methods suffered from a constant bias, as we do not observe a decreasing relative error as the distance increases. `BAP` and `NOUR` present a stable relative error for all distances, i.e., with approximately 0.3 % of standard deviation. This indicates that the estimation suffered only from a scale error, which will be observed later and addressed in section 4.3.3. One could thus re-scale the poses to provide a precise and accurate estimation. We cannot draw any conclusion for the other methods since the variations do not follow any obvious pattern.

Finally, our model differs from the model of Noury *et al.* [166] by modeling the micro-lens focal lengths. Comparing those two models, the mean error as well as the standard deviation is smaller with our method. The inclusion of the micro-lens focal lengths in the camera model improves the precision and accuracy, and enables to generalize to several configurations. Dealing with different intrinsics which produce different extrinsics is not satisfactory when using the multi-focus plenoptic camera. In contrast, our model is able to manage all micro-lens types simultaneously, and proves to be stable across various configurations and working distances.

3.4.2.4 Relative blur calibration

We calibrate the blur proportionality coefficient κ for the three datasets using our BAP features. Figure 3.34 presents two windows extracted around BAP features of different types from the same cluster, showing different amount of blur. The target image to be equally-defocused according to our model is shown before, (b), and after, (c), blur addition. The estimated PSF of the relative blur is given in (d). The optimized blur proportionality coefficients κ are reported in Table 3.4. Theoretically, the parameter should be the same for all three datasets. Empirically this observation is validated for `R12-A,B` and `C`. Estimated κ for `R12-D` and `R12-E` are higher. This

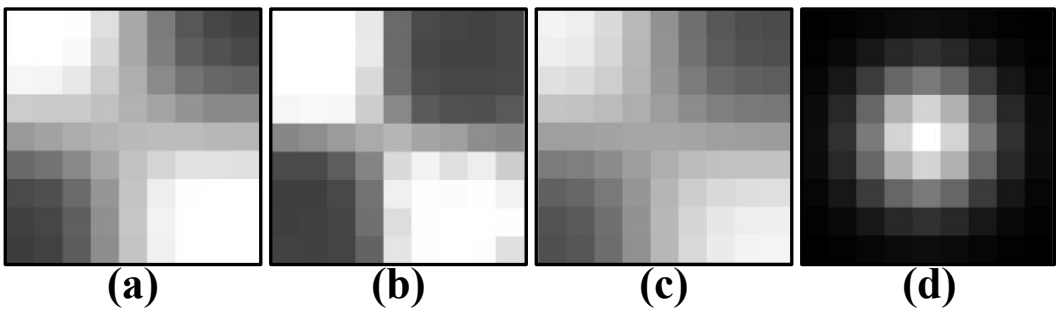


Figure 3.34: Example of micro-images before and after being equally-defocused. **(a)** Reference image with highest amount of blur. **(b)** Target image to be equally-defocused. **(c)** Target image with additional blur. **(d)** Estimated point-spread function (PSF).

is because the micro-lenses focal lengths in R12-D and R12-E are slightly bigger than in R12-A,B,C. Analytically, this difference generates a smaller amount of relative blur, and thus a higher estimate of κ to match the observed blur in image space. In other words, κ compensates for the slight differences in $f^{(i)}$ estimates. Therefore, κ should be calibrated for each dataset.

3.4.3 Ablation study

To evaluate the influence of each parameter of the camera model, we present an ablation study of some of them. We focus the analysis on distortion coefficients (Q_1, Q_2, Q_3, P_1, and P_2), on some degrees of freedom of the MLA, especially its tilt with respect to the sensor (θ_x, θ_y), and the pitch between micro-lenses (Δ_μ). All combinations of the parameters have been tested, resulting in eight configurations. For each configuration and on each dataset of R12-A,B,C: first, we calibrate the camera intrinsic parameters; second, we evaluate the model using the RMSE of the BAP reprojection; and finally, we quantitatively estimate the relative translation error on the evaluation dataset. Each configuration has been initialized with the same intrinsic parameters, and used the same observations for all processes. Results are reported in Table 3.12. The first column is the configuration number. The Tilt column indicates if we keep (✓) or remove (×) the parameters θ_x and θ_y. The Pitch column stands for the parameter Δ_μ, and the column Dist for the distortion parameters Q_1, Q_2, Q_3, P_1, and P_2. The reprojection error ε_{all} is given by its RMSE, and the relative translation error ε_z is expressed in percent with respect to the ground truth displacement. The configuration 1 is our reference, corresponding to

Table 3.12: Ablation study of some camera parameters. For each dataset, the reprojection error ε_{all}, computed using the RMSE along with the relative translation error ε_z, expressed in %, are reported. The symbol ✓(*resp.,* ×) indicates if we keep (*resp.,* remove) the considered parameters.

	Tilt	Pitch	Dist	R12-A		R12-B		R12-C	
				ε_{all}	ε_z	ε_{all}	ε_z	ε_{all}	ε_z
1	✓	✓	✓	**0.860**	**3.23**	**0.698**	**3.15**	0.739	2.31
2	✓	✓	×	0.866	3.34	0.700	3.20	**0.737**	3.18
3	✓	×	✓	0.884	3.88	0.755	3.21	0.770	**2.00**
4	✓	×	×	0.891	3.98	0.752	3.24	0.773	3.13
5	×	✓	✓	0.865	3.48	0.784	**3.15**	0.760	2.89
6	×	✓	×	0.864	3.58	0.716	3.16	0.749	3.04
7	×	×	✓	-	-	-	-	-	-
8	×	×	×	-	-	-	-	-	-

the complete model. The optimized parameters are close to the ones from Tables 3.7-3.9, i.e., with less than 1 % of variation, for all converging configurations and for all evaluated datasets.

First, distortion does not impact the reprojection error of the model. Considering the pairs of configurations $(1, 2)$, $(3, 4)$, and $(5, 6)$, the errors are similar with or without distortion, indicating that our camera does not suffer from lateral distortion. This is due to the relatively large main lens focal length. Nevertheless, distortion may have a role to play in case of shorter focal length.

Second, removing the rotations of the MLA does not improve nor worsen the reprojection error and the pose estimation. When keeping the tilt but freezing the pitch, the model is able to converge. The tilt, in combination with other factors (such as a slight decrease of the main lens focal length), compensates for the error introduced by the approximate value of the pitch. In contrast, configurations 7 and 8 do not converge to a solution, showing that when removing both the tilt and the pitch of the MLA, the model is not constrained enough, and the reprojection error cannot be minimized, resulting in a failure.

Finally, when freezing the pitch to its initial value, the positional part of the reprojection error increases. It is especially the case for dataset `R12-A`, where the reported errors in Table 3.12 are the highest of all configurations. This confirms our previous observation that the deviation of the micro-image centers and their optical centers does not satisfy an orthographic projection between the MIA and the MLA. The pitch should be taken into account, on the one hand to improve the precision of the model, and on the other hand not to hinder the optimization process.

3.5 Application to depth of field profiling

3.5.1 Extended depth of field

From calibrated camera parameters, we can compute the depth of field (DoF) of each micro-lens type and the *blur profile* – the blur radii as function of the object distance –, in order to profile the plenoptic camera. The analysis can be done with respect to the MLA frame, and then extended to object space by back-projection of the z-component. A point at a distance a from MLA is projected back into object space at a distance a' according to the thin-lens equation through the main lens, such as

$$a' = \frac{(D - a) \cdot F}{(D - a) - F}. \tag{3.78}$$

Let r_0 be the minimal acceptable radius of the circle of confusion (CoC) given by Eq. (3.26). For a micro-lens of type (i), the focus plane distance is given by

$$a_0^{(i)} = \left(\frac{1}{f^{(i)}} - \frac{1}{d}\right)^{-1} = \frac{df^{(i)}}{d - f^{(i)}}. \tag{3.79}$$

With Δ_i the micro-lens aperture, we derive then the *far* a_+ and *near* a_- focus planes distances:

$$\begin{cases} a_+^{(i)} = \dfrac{d \cdot \Delta_i \cdot a_0^{(i)}}{\Delta_i \cdot f^{(i)} - 2r_0\left(a_0^{(i)} - f^{(i)}\right)} & \text{[far]} \\ a_-^{(i)} = \dfrac{d \cdot \Delta_i \cdot a_0^{(i)}}{\Delta_i \cdot f^{(i)} + 2r_0\left(a_0^{(i)} - f^{(i)}\right)} & \text{[near].} \end{cases} \tag{3.80}$$

The DoF of a micro-lens of type (i) is computed as the distance between the near and far focus planes, such as

$$\mathrm{DOF}^{(i)} = \left|a_+^{(i)}\right| - \left|a_-^{(i)}\right| = \frac{\Delta_i \cdot f^{(i)} \cdot a_0^{(i)} \cdot 2r_0\left(a_0^{(i)} - f^{(i)}\right)}{\left(\Delta_i \cdot f^{(i)}\right)^2 - 4r_0^2\left(a_0^{(i)} - f^{(i)}\right)^2}. \tag{3.81}$$

Note that to fully exploit the combined extended DoFs without gaps, the micro-lenses DoFs should either just touch or slightly overlap [5]. Finally, under this consideration, the total DoF of the plenoptic camera in MLA space is computed using the micro-lenses DoFs as

$$\mathrm{DOF} = \max_i \left\{\left|a_+^{(i)}\right|\right\} - \min_i \left\{\left|a_-^{(i)}\right|\right\}. \tag{3.82}$$

3.5.2 Blur profiles

Using the parameters obtained from our calibration, we plot the *blur profile* of the camera, i.e., the evolution of the blur radius with respect to depth for each micro-lens type along with its corresponding DoF. Figure 3.36 shows the blur profiles obtained for three focus distance configurations, `R12-A,B,C`, with their DoFs expressed in mm. The blur radius is expressed in pixel and is given for each type, in *red* for type (1), in *green* for type (2) and in *blue* for type (3). Distances are given in object space in mm with their corresponding virtual depth on a secondary x-axis, spanning from $v = 1$ to 15, except for the configuration $h = \infty$ where we cropped just after the farthest focal plane. In MLA space, the profiles have the same behavior for all focus distances, as it only depends on the MLA parameters which are common to all configurations.

First, the horizontal dashed line represents the radius of the minimal acceptable circle of confusion ρ_0. In our case, at a wavelength of 750 nm, the radius of the smallest diffraction-limited spot is $r^* = 2.4$ µm which is less than half the pixel

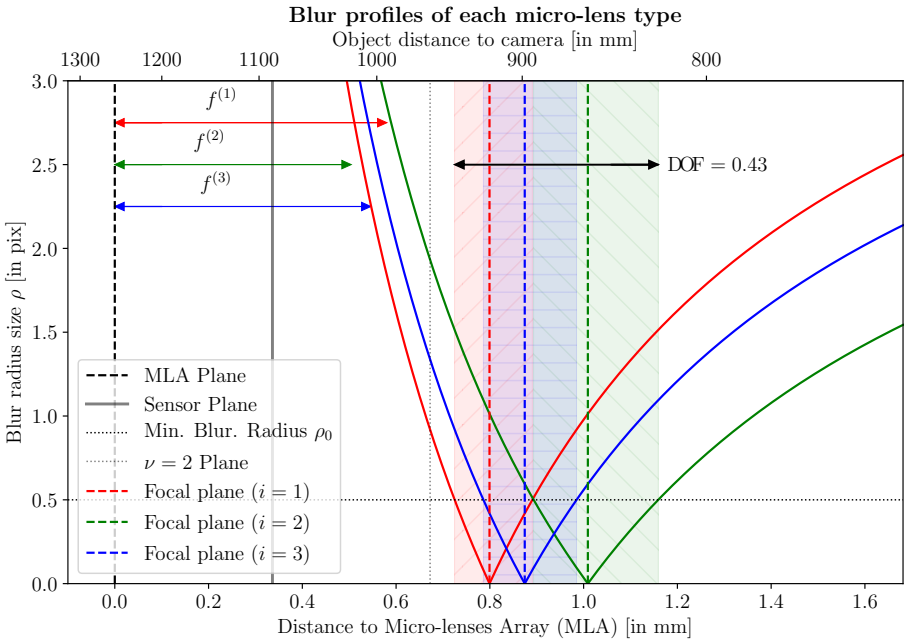


Figure 3.35: Blur profiles for each micro-lens type in MLA space, for the dataset `R12-B`, with $h = 1000$ mm.

size. We then choose $r_0 = s/2$, i.e., $\rho_0 = 1/2$. Despite not illustrated in the figure, the blur radius grows exponentially when getting closer to the plane $v = 0$. Once this limit is exceeded, the blur decreases and converges to a constant value of approximately 6 pixel. This happens for more distant objects when points are projected in front of MLA implying a negative virtual depth. This is the case for $h = 450$ and $h = 1000$ mm, but not for $h = \infty$, as the points were never projected closer than $v = 2$. In the working distance range, the blur does not exceed 5 pixel and grows when points are closer to the camera.

Secondly, we can use the DoF to select the range of working distances where the blur is not noticeable. The DoF increases in object space as the focus distance increases. As reported on the figures: for `R12-A`, the DoF is of 14.44 mm; for `R12-B` of 120 mm; and finally, for `R12-C`, the total DoF is of 223 m. In MLA space the total DoF is constant and spans from $v \approx 2.15$ to 3.45 (as illustrated by Figure 3.35). As expected, the DoFs overlap. In particular, the DoF of the type (3) micro-lens is entirely included in the other two, whereas the DoFs of the type (1) and (2) just touch. Within the total DoF, a point can then be seen focused in two micro-images of different types simultaneously, which eases the matching problem between views.

Finally, we can easily identify the distance limits at which the point will not be in the DoF anymore nor be projected on multiple micro-images, i.e., corresponding to virtual distances $|v| < 2$. At these distances, disparity cannot be computed in image space, and no depth estimation can be performed. Such estimation can also be hindered by the resolution in virtual space compared to the resolution in object

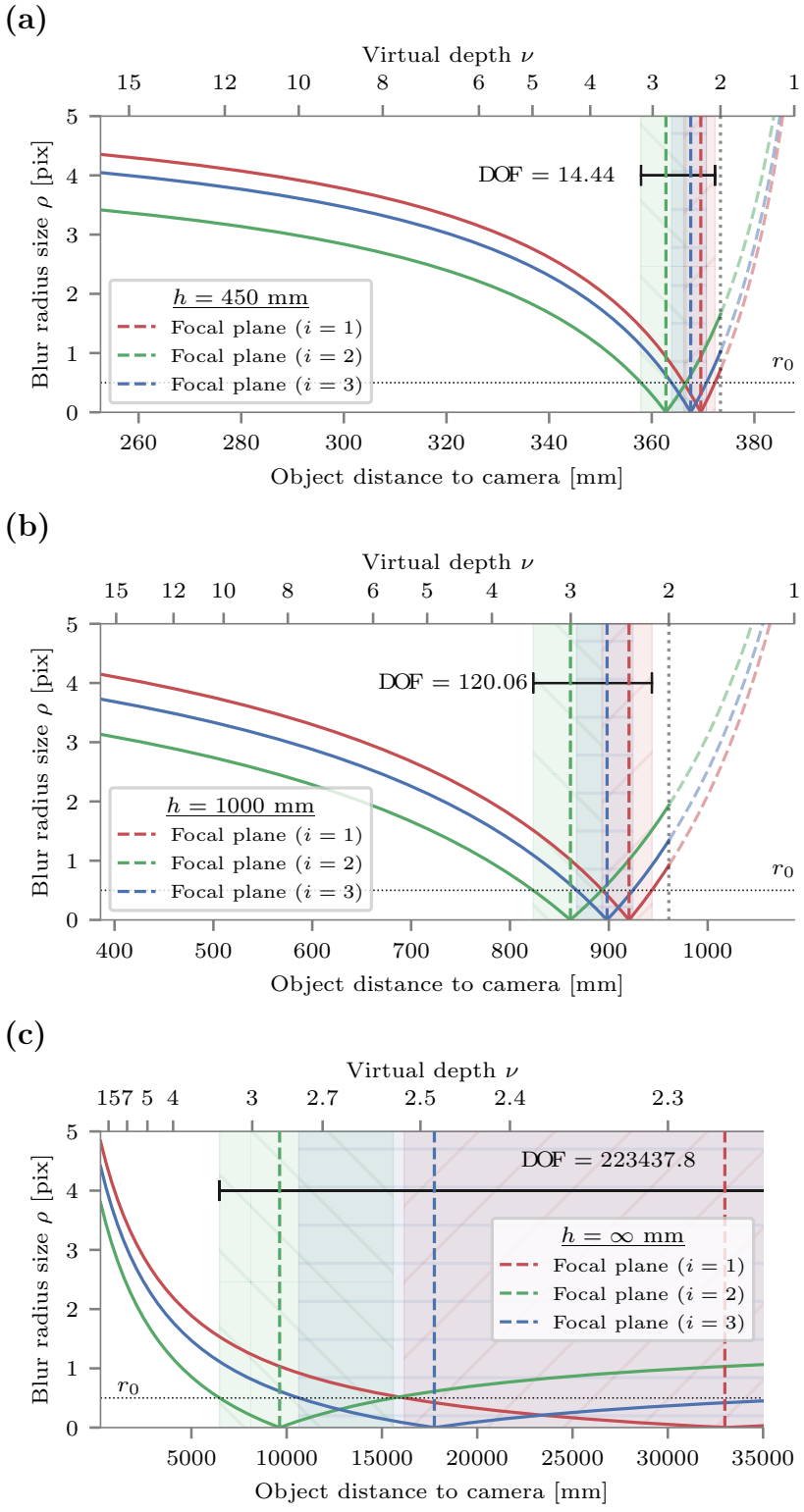


Figure 3.36: Blur profiles, including each micro-lens type, in object space, at different focus distances: **(a)** $h = 450$ mm; **(b)** $h = 1000$ mm; and **(c)** $h = \infty$ mm. Focal planes and DoFs are illustrated for each type. The blur radius is expressed in pixel as function of the object distance to the camera in mm. Corresponding virtual depth is reported on the secondary x-axis. Points projected closer than the plane $\upsilon = 2$ plane can't be used by stereoscopic algorithm. Points at closer distance tend to have a constant blur radius. Points within the DoFs of each micro-lens can't be used for blur analysis.

space as disparity is inversely proportional to virtual depth. For instance, for close objects, points will be projected on more micro-images but with a low disparity. So the profiles can be used to efficiently characterize the range of distances according to the desired application.

Furthermore, once the MLA parameters are available, we can simulate an approximate blur profile for the desired focus distance h with the desired main lens focal length F by updating the value of D using Eq. (3.56) and Eq. (3.28).

Conclusion

In this chapter, we addressed the problem of plenoptic cameras *calibration*, to answer the question "*How can we link world space information to the image space information?*". To calibrate a plenoptic camera, state-of-the-art methods rely on simplifying hypotheses, on reconstructed data or require separate calibration processes to take into account the multi-focus configuration. Taking advantage of blur information, we proposed: 1) a more complete plenoptic camera model with the introduction of a new BAP feature that explicitly models the defocus blur; this new feature is exploited in our calibration process based on non-linear optimization of reprojection errors; 2) a new relative blur calibration to fill the gap between the physical and geometric blur, which enables us to fully exploit blur in image space; and 3) a profiling of the plenoptic camera and its extended depth of field.

Our camera model is applicable to the multi-focus plenoptic camera (both in Galilean and Keplerian configuration), as well as to the single-focus and unfocused plenoptic camera. In case of the `Raytrix` multi-focus camera, our ablation study shows that main lens distortions and MLA's tilt can be omitted without hindering the calibration process nor the pose estimation. The study also indicates that explicitly including the pitch of the micro-lenses in the model improves the results. In addition, our calibration methods are validated by quantitative evaluations in controlled environment on real-world data. Our method provides strong initial intrinsics during the pre-calibration step, and coherent optimized camera parameters for all evaluated configurations. It shows a low and stable relative translation error across all the datasets. Thus, our model generalizes to various configurations.

Precise and accurate parameters of the camera model can thus be leveraged for applications such as depth estimation. The critical step of calibration is required if we want to achieve metric results. In the next chapter, we will use blur information in complement to disparity to improve metric depth estimation, i.e., to answer the more complex question "*How can we link image space information to world space information?*".

Chapter 4

Depth estimation with plenoptic cameras

Introduction

This chapter covers the problem of depth estimation from a single image acquired with plenoptic cameras. It aims to answer the question "*How can we link image space information to world space information?*". First, we review the existing methods for depth estimation based on light-field data. We see that most of them are working with SAIs or EPIs which is prone to error as depth is usually required to reconstruct the light-field or the SAIs, especially in the focused plenoptic camera case. To overcome this issue, algorithms can work directly with the raw plenoptic images, at micro-images level. However, usually only micro-images with the smallest amount of blur are used, or alternatively, specific patterns are designed to exploit the information [185]–[187]. In contrast, using our camera model, we show how to relate the camera parameters to the amount of blur in the image, and how all information can be used simultaneously, without distinction between types of micro-lenses. We propose then to leverage blur information where it was previously considered as a drawback. Second, we explain how we link the disparity in image space to the defocus blur information. Indeed defocus cues are complementary to correspondence cues, and can improve the quality of depth estimation [188]. Third, we introduce and detail our blur aware depth estimation (BLADE) framework as well as the depth calibration process. Indeed, we will see that the inverse projection has a depth scaling error. We present then a methodology to measure this error and to correct it. Finally, our experimental setups are presented and our results are given and discussed for the relative depth estimation setup and for the real-world 3D complex scenes with ground truths acquired with a 3D lidar scanner.

Contributions

We propose a new metric depth estimation algorithm using only raw images from plenoptic cameras. It is especially suited for the multi-focus configuration where several micro-lenses with different focal lengths are used. Our contributions are three-fold. First, we introduce a metric depth estimation framework for plenoptic cameras, named BLADE, leveraging both spatially-variant blur and disparity cues between micro-images. It is based on area matching techniques to estimate a raw depth map $\mathcal{D}$ directly from raw plenoptic images. Two variations are considered: 1) coarse estimation, i.e., one depth per micro-image; and 2) refined estimation, i.e., one depth per pixel. Second, we include in our inverse model a depth scaling correction as we are able to measure and characterize this error. We give a methodology to correct it in a post-calibration process. Finally, we present a new dataset of 3D real-world scenes with ground truths acquired with a 3D lidar scanner, and a methodology to calibrate the extrinsic parameters.

4.1 Background

In this section, we briefly review classical paradigms for depth estimation and how they apply to plenoptic imaging.

4.1.1 Depth from stereo

The stereo correspondence problem is the process of ascertaining which parts of one image correspond to which parts of another image. Correspondence is a fundamental problem in computer vision. It can be generalized to the N-view correspondence problem. Once solved and with the cameras parameters, it can be used to estimate the 3D depth information of a scene. This is achieved by determining the disparity of matched pixels between the stereo viewpoint images [189] (either using area-based or feature-based approaches). This problem relies on the following constraints hypotheses:

Similarity Constraint both projections of the same 3D object should have similar properties or attributes (e.g., shapes, colors, sizes, vertices);

Geometry Constraint a pixel from one view must match against a pixel on the other view onto the same epipolar plane according to camera geometry when images have been rectified;

Uniqueness Constraint a feature from the reference view image has one and only one feature related to it on the target view image.

However, in the case of the plenoptic camera, only the *geometry constraint* is satisfied as micro-lenses are supposed parallel to the image plane, no rectification is needed, and the projection follows epipolar geometry. The *uniqueness constraint* is violated when an occlusion occurs, but the problem might be resolved using angular coherence as a point is usually projected onto more than one micro-image. Furthermore, the *similarity constraint* is violated due to the difference of focus between two micro-images types. To satisfy the constraint, one might only compare images of the same type, or restrict the working range to the DoF of the camera (as at least two micro-images are in-focus simultaneously). Alternatively, higher level features can be used. On the other hand, the defocus being one of the main sources of uncertainty, we can include blur information within the depth estimation process. Our framework addresses the latter option as we want to exploit all the available information.

4.1.1.1 Triangulation

A 3D point can be estimated if it is observed by at least two cameras with known intrinsic and extrinsic parameters. Triangulation is the process of computing the intersection of light rays emanating from observations of the 3D point. In practice, due to the data noise (intrinsic parameters, camera poses, features detection, etc.), light rays might not intersect. In case of two cameras, the result of triangulation is usually set as the equidistant point to both rays, as illustrated by Figure 4.1. In order to be robust and precise in numerical computation, triangulation can be

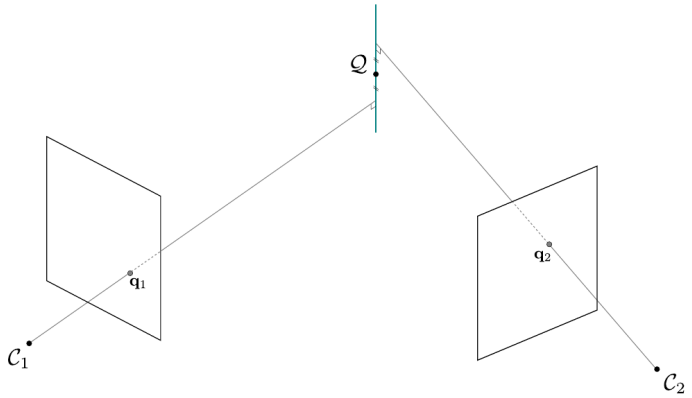


Figure 4.1: 3D point triangulation.

generalized to N cameras. For instance, in case of three cameras, we can compute three distinct triangulations from each pair. The final result is the barycenters of these three points. In case of lenslet-based plenoptic cameras, each micro-lens can be considered as a single camera, and with known calibration, triangulation can be applied to the micro-images.

4.1.1.2 Feature-based matching techniques

Comprehensive review on features detection and description can be found in [190]–[192]. In context of plenoptic imaging, use of interest points matched between micro-images has also been investigated by Konz *et al.* [193] to estimate virtual depth by triangulation. However, the accuracy of the estimation is limited, and the number of features too low to obtain a proper depth map. Area-based matching techniques seem then best suited to the plenoptic case.

4.1.1.3 Area-based matching techniques

Area-based techniques solve the matching problem by using the intensity patterns of the neighborhood of a reference pixel to determine its correlation. They estimate the correlation between the distribution of disparity for each pixel in an image using

a window centered at the reference pixel along the epipolar lines. The effectiveness of these techniques depends largely on the window size taken: smaller windows are not discriminant enough, and larger windows increase the computational cost. For rectified image pairs $(\mathcal{I}^*, \mathcal{I})$, we can cite the following measures:

Sum of absolute differences (SAD) [194] measures the similarity of a pixel from a reference image in the target image by summing the intensity values inside a window including the discrete pixel from the target image and finding the same best match of that block in the reference image. The error is given by

$$\varepsilon_{\text{SAD}}(x, y, \delta) = \sum_{(i,j)\in\mathcal{N}(x,y)} \left|\mathcal{I}^*(i, j) - \mathcal{I}(i + \delta, j)\right|, \tag{4.1}$$

where $\mathcal{I}^*$ is the reference image to be compared to the target image $\mathcal{I}$ at disparity δ, for pixels within the neighborhood $\mathcal{N}$ around the considered pixel at (x, y).

Sum of squared differences (SSD) [195] measures the similarity of a pixel from a reference image in the target image by summing the squared intensity values of window including the discrete pixel from the target image and finding the same best match of that block in the reference image. The error is given by

$$\varepsilon_{\text{SSD}}(x, y, \delta) = \sum_{(i,j)\in\mathcal{N}(x,y)} \left(\mathcal{I}^*(i, j) - \mathcal{I}(i + \delta, j)\right)^2, \tag{4.2}$$

where $\mathcal{I}^*$ is the reference image to be compared to the target image $\mathcal{I}$ at disparity δ, for pixels within the neighborhood $\mathcal{N}$ around the considered pixel at (x, y).

Birchfield-Tomasi measure (BT) [196] measures the dissimilarity of a pixel from a reference image in the target image by linearly estimating the interpolated values of a window match and its nearest neighbors pixels. The measure is insensitive to image sampling. The error is given by

$$\varepsilon_{\text{BT}}(x, y, \delta) = \min \left\{ \begin{array}{l} \min\limits_{x-\frac{1}{2}\le i\le x+\frac{1}{2}} \left|\mathcal{I}^*(x, y) - \hat{\mathcal{I}}(i + \delta, y)\right| \\ \min\limits_{x-\frac{1}{2}\le i\le x+\frac{1}{2}} \left|\mathcal{I}(x + \delta, y) - \hat{\mathcal{I}}^*(i, y)\right| \end{array} \right\}, \tag{4.3}$$

where $\mathcal{I}^*$ is the reference image ($\hat{\mathcal{I}}^*$ is its interpolated intensity) to be compared to the target image $\mathcal{I}$ (its interpolated intensity is given by $\hat{\mathcal{I}}$) at disparity δ.

Normalized cross-correlation (NCC) [195] measures the similarity of a pixel from a reference image in the target image using the Cauchy-Schwarz inequality. NCC is computationally more expensive compared to the SAD and SSD

techniques due to numerous multiplications, division and square root operations, but is more robust. The error is given by

$$\varepsilon_{\text{NCC}}(x,y) = \frac{\sum_{(i,j)\in\mathcal{N}(x,y)} \sum \left[\mathcal{I}^*(i,j) \cdot \mathcal{I}(x+i,y+j)\right]}{\left[\sum_{(i,j)\in\mathcal{N}(x,y)} \sum \left[\mathcal{I}(x+i,x+j)\right]^2\right]^{\frac{1}{2}}}, \tag{4.4}$$

where $\mathcal{I}^*$ is the reference image to be compared to the target image $\mathcal{I}$.

Census transform (CT) [197] reduces the image intensity composition of an image data into binary intensity values depending on the value of the center pixel. It is insensitive to global radiometric variations, but is highly dependent on the center pixels and on the window size, which could result in high computationally cost. Similarity between images is determined by comparing the values of the census transform for corresponding pixels, using the Hamming distance.

A thorough analysis of cost aggregation is conducted in [198] to analyze their performance on light-field depth estimation. Although less discriminant, the SAD measure can still be considered as a cost-efficient strong solution for area-matching.

4.1.2 Depth from focus/defocus

The depth from focus/defocus approach aims to estimate the spatially variant spread parameter of the blur kernel, by acquiring two images of the same scene with different camera settings [132], [133], [199], [200]. The blur radius is linked to the inverse distance, and once the blur is estimated, depth can be retrieved. The spread parameter is usually estimated in the frequency domain or in the spatial domain [177]. Depth from focus/defocus works better on short ranges and can thus be seen as a solution for these distances. Indeed, depth from stereo methods are less effective in a short range due to part of the scene being not visible by both cameras. However, the main hypothesis is that the two images represent the same scene viewed from the same point of view and with the same view angle, which is not the case with a plenoptic camera. Even if the multi-focal lengths provide defocus cues for depth estimation, the micro-images have to be matched according to the parallax.

4.2 Related work

In this section, we will analyze the different approaches to estimate *depth from light-field* from the state-of-the-art. Most light-field depth estimation processes

operate in two steps: 1) initial depth map estimation from SAIs or EPIs, and 2) depth refinement with global methods. An overview and taxonomy of dense light-field depth estimation algorithms is available in [114], but mostly including methods working on reconstructed images such as SAIs or EPIs.

4.2.1 Depth from sub-aperture images (SAIs)

One category of approaches estimates depth from reconstructed SAIs. Perez Nava and Luke [201] proposed a framework that simultaneously estimate an all in-focus image along with the depth map, based on focal stack analysis for the focused plenoptic camera. In [62], traditional multi-view stereo methods are applied to reconstruct both scene depth and its super-resolved texture in a Bayesian framework. To overcome the issue of needing depth to generate super-resolved image, they proposed an iterative process by applying directly onto micro-images an antialiasing filter, and refining the depth estimate based upon the current depth map.

Correspondence and defocus cues have been analyzed by Kim *et al.* [202] to select reliable pixels for depth estimation using a cost volume reconstructed from the light-field. Their method showed stable depth estimation but requires high spatio-angular resolution and uses SAIs with a significantly larger baseline than the lenslet-based light-field image.

Jeon *et al.* [203] presented a depth map estimation algorithm using a multi-label optimization of a cost volume for lenslet-based light-field image. To achieve sub-pixel disparity estimation, SAIs were directly shifted using a phase shift theorem. The estimated depth map is then iteratively refined to obtain a continuous disparity map. They improved their depth estimation in [204] by combining different matching costs and learned to automatically determine which combination performs better on the given input. Several aggregation costs have also been tested and evaluated by Williem and Park [198].

Built upon the work of Tao *et al.* [188], Wang *et al.* [11] conducted depth estimation by treating occlusion explicitly in the photo-consistency model using the reconstructed central view. Using not only the depth map in the central view but also view-wise depth maps significantly improves the performance of depth estimation, as stressed out by Peng *et al.* [205]. To better accounts for correlation and dependencies within angular patches and spatial images, Zhang *et al.* [206] proposed a two-step light-field depth estimation based on graph spectral analysis.

In another direction, recent work of Anisimov *et al.* [207] aimed at reducing computational cost by leveraging a semi-global matching strategy, instead of focusing on improving depth estimation. Their method is based on pixel matching in SAIs

with similarity measurement based on the Census transform (CT) with Hamming distance, for estimation of a dense depth map.

Another category of methods aims to leverage the light-field multi-views structure to extract features to be matched. Several descriptors have been proposed such as LiFF [208] built upon SIFT, the binary descriptor introduced by Alain and Smolic [209] built upon BOOM, or FDL-HSIFT [210] built upon Harris and SIFT in the scale-disparity space.

All the previous methods operate either on the light-field or on reconstructed SAIs. It easily available with a camera-array setup, sequential acquisition, or an unfocused plenoptic camera. This is not the case for the focused plenoptic camera. Indeed, the latter setup leads to an ill-posed problem because depth information is required to reconstruct the light-field or the SAIs.

4.2.2 Depth from epipolar plane images (EPIs)

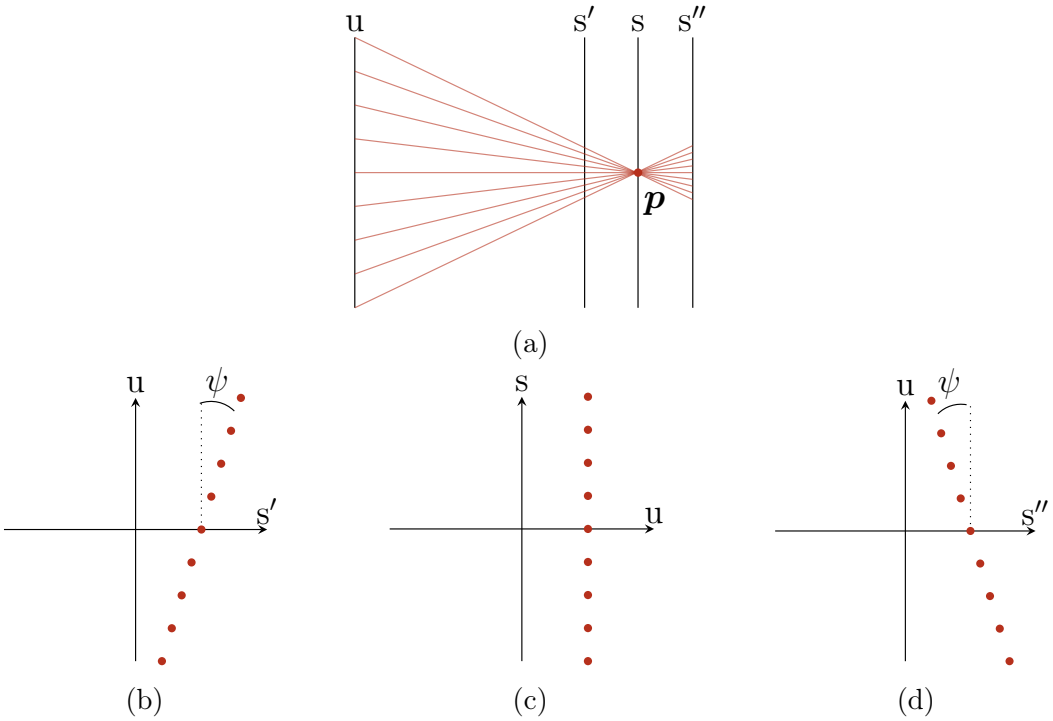


Figure 4.2: Distance computation from EPI analysis. The angle ψ is function of the distance of the point $\boldsymbol{p}$ with respect to the different planes of focus s, s′ and s″ (a). In the EPI (c), we have $\psi = 0$ when $\boldsymbol{p}$ is located on the s plane. However, when $\boldsymbol{p}$ is located behind (b), i.e., on the s′ plane, and after (d), i.e., on the s″ plane, the angle ψ is non-null.

As seen in section 2.3.2, EPI usually represents a 2D slice (in the spatial and angular dimensions) of the 4D light-field. From analysis of variations within this representation, we can infer depth information, as shown in Figure 4.2. This structure was first analyzed by Bolles *et al.* [49] to retrieve depth from lines detected in the EPI, as the slope is inversely proportional to depth.

For the focused plenoptic camera, Wanner *et al.* [211] proposed an algorithm to generate EPI representation. To overcome the necessity to estimate depth to generate EPI, they proposed to compute the best patch size by local minimization over all possible rendered focal images. They computed the full depth of field view for each lens type independently and then applied a merging algorithm, to include the multi-focal aspect. Wanner and Goldluecke [212] used the latter representation by proposing a globally consistent framework using structure tensors to estimate the directions of feature pixels in the 2D EPI. In the work of Yu *et al.* [213], geometric structures of 3D lines in ray space extracted from EPI were explored. They encoded the line constraints to further improve the reconstruction quality. A method was proposed by Tosic and Berkner [214] to detect ray geometry in EPIs based on light field scale and depth (Lisad) space transform. From rays information, they computed a depth at each pixel by converting angle to depth.

An algorithm that computes dense depth estimation by combining both defocus and correspondence depth cues was introduced by Tao *et al.* [188]. They analyzed both vertical and horizontal EPIs, as the first one informs about correspondence, whereas the second gives defocus information. They latter included shading as a third clue to improve their depth estimation [12]. Xu [215] implemented a three-step depth estimation using EPIs. First, they estimated slopes in the vertical and the horizontal slices, and converted them to disparities. Second, both disparity maps were fused, and finally globally refined using a Markov random field (MRF).

Latter methods are vulnerable to occlusion as it generates inconsistencies in the EPIs. Chen *et al.* [216] explicitly tackled this issue by presenting a new light-field stereo matching algorithm that is capable of handling occlusion based on analysis of the angular statistics of the light-field. Their method performed well in the absence of noise. More recently, Zhang *et al.* [217] introduced a spinning parallelogram operator (SPO) to locate lines and calculate their orientations in an EPI for local robust depth estimation. According to the authors, SPO has been demonstrated to be insensitive to occlusion, noise, spatial aliasing, or limited angular resolution. A new framework using SPO was developed by Sheng *et al.* [218] to locate lines in multi-orientation EPIs. Inconsistency of labeling within orientations allows to detect occlusion boundaries'. Depth map is then computed with a global optimization taking into account depth estimates and occlusions.

Similarly to SAIs-based methods, the EPI representation can easily be retrieved for unfocused plenoptic cameras, but needs prior depth to be generated from focused plenoptic cameras.

4.2.3 Depth from learning

Deep learning methods have also been applied on light-field images, in particular in context of super-resolution. The first deep convolutional neural network (CNN) that jointly optimized angular and spatial super-resolution images from a pair of SAIs was proposed by Yoon *et al.* [65]. Generated SAIs were then used in a stereo matching-based depth estimation built-upon the method of Jeon *et al.* [203]. Ma *et al.* [219] introduced an end-to-end network using all SAIs allowing to capture both local and global features to generate a disparity map. It performed well in texture-less areas but poorly to preserve details. To reduce the amount of input data, Shin *et al.* [220] developed a multi-stream fully CNN using only SAIs stacked in four angular directions to produce a disparity map. Liu *et al.* [221] presented a three-part neural network architecture, where the first part processes a focal stack, the second part is the architecture of Shin *et al.* [220], and finally both outputs are compared in a third part. It allows to take into account both parallax and ambiguity cues.
Recently, a new depth estimation based on unsupervised learning was proposed in [222] to overcome the necessity of having depth maps as ground truth and to reduce the gap between simulated and real data. Contrarily to previous methods, Leistner *et al.* [223] proposed a neural network which aims at estimating depth for wide-baseline light-field which had not been addressed yet.

Using EPI representation, Johannsen *et al.* [224] proposed a novel approach for depth estimation based on a learned dictionary which codes for disparity from EPI. Heber *et al.* [225] introduced a U-shaped fully CNN with skipped connections where inputs are EPI representations and output is a disparity map. Their method has the advantage of having a very low computational time compared to existing solutions. Recently, Li *et al.* [226] proposed a pseudo-Siamese neural network to estimate depth at each pixel, taking as input the vertical and the horizontal EPI at this location. Instead of explicitly including vision cues, Huang [227] proposed to model the light-field matching problem using an empirical Bayesian framework which better generalizes to different light-fields (dense, sparse, color, gray-scaled, etc.) to achieve better depth quality.

To the best of our knowledge, no learning-based methods are able to directly operate on raw plenoptic images, and therefore can be applied to focused plenoptic cameras.

4.2.4 Depth from raw images

To overcome the issues related to reconstruction of the SAIs or EPIs, several methods work directly with the raw images. This is particularly suited for the focused plenoptic camera, as each micro-image captures more spatial information than its unfocused counterpart.

With the arrival of commercial focused plenoptic cameras, Perwaß and Wietzke [5] proposed a methodology to estimate depth directly from raw images. Their method is based on triangulation from micro-images views employing a correlation technique just as in standard stereo matching approaches. But, it required contrasted micro-images and sharp micro-images. If the camera is calibrated and once each pixel has a depth estimate, sparse metric depth can be retrieved. Custodio [228] presented an automatic method to estimate the depth of a scene based on multi-view geometry and ray back-tracing from detected salient points. Use of points of interest matched between micro-images (SURF, SIFT and Harris) has also been investigated by Konz *et al.* [193] to estimate virtual depth by triangulation. However, the accuracy of the estimation is limited, and the number of features too low to obtain a proper depth map.

Noury [127] conducted metric depth estimation per micro-image. Their method is inspired from standard dense stereo matching techniques [189] but applied to micro-images from the raw plenoptic image. It relies on a minimization process of the dense reprojection error of the reconstructed neighbors micro-images given a depth hypothesis following the projection model of the camera.

Depth estimation for the multi-focus plenoptic camera has been explicitly considered by Fleischmann and Koch [185]. Their method, based on one depth estimation per micro-image, operates by regularizing a cost volume computed from a similarity measure between micro-images at different disparity hypotheses. Disparities are then converted to virtual depths, according to the MLA parameters. To take into account the varying amount of defocus blur between micro-images of different types, they developed an adaptive strategy to select only certain candidates micro-images. Similar to the previous work, Ferreira and Goncalves [186] used salient points detected with SIFT to select micro-images in which a search along epipolar line is performed. A specific lens selection scheme to improve robustness is proposed. Finally, matched points are back-projected into virtual space to form a point cloud. Virtual points are then reprojected and an average depth is attributed for every micro-image. Palmieri and Koch [187] have also addressed lens selection strategies. An other depth estimation for the multi-focus plenoptic camera is proposed by Cunha *et al.* [229]. The method operates by first detecting edges in micro-images, which are second matched with neighbors micro-images. Finally, matched points are triangulated into

virtual space, and then reprojected into metric space with calibration parameters available. It addressed the issue of the different amount of blur during the matching by proposing a switching mechanism between intensity and local phase quantization (LPQ) domains.

Zeller *et al.* [230] introduced the first probabilistic depth estimation from raw plenoptic images obtained from a focused plenoptic camera. They addressed depth estimation as a multi-view stereo problem. For each pixel having a sufficient gradient, virtual depth hypothesis is obtained by finding correspondences along epipolar lines in neighbors micro-images with local intensity error minimization. Multiples hypotheses are merged in a Kalman-like fashion, allowing to associate a variance to the estimation. To deal with the multi-focus aspect, they incorporated a term modeling the focus uncertainty.

All previous solutions from the state-of-the-art when working on raw images considered blur as a drawback and designed specific strategies to select micro-images. On the other hand, leveraging our new camera model presented in chapter 3, we can explicitly use the defocus information in the depth estimation process, taking into account both correspondence and defocus cues.

4.3 Proposed depth estimation method (BLADE)

In this section, we present our contribution to depth estimation using raw plenoptic images and taking both correspondence and defocus cues into consideration. Our method is based on the minimization of a newly introduced cost function leveraging both defocus and a disparity hypotheses. An overview of the process of computing the error is given in Figure 4.3. When matching micro-images contents of different types, we are in a *defocus stereo configuration*. As highlighted in [184], while estimating depth from a defocus stereo configuration, both spatially-variant blur and disparity provide the inference for depth information [231], [232]. Schechner and Kiryati [232] and Vaish *et al.* [233] extensively discussed the advantages and disadvantages of each cue. Establishing the visual correspondence across two images must take both disparity and blur into account. The goal is then to improve disparity estimation for defocus stereo images via compensating the mismatch of focus and integrating both correspondence and defocus cues. But first, we need to link the disparity, using the virtual depth, to the amount of blur generated in the micro-images.

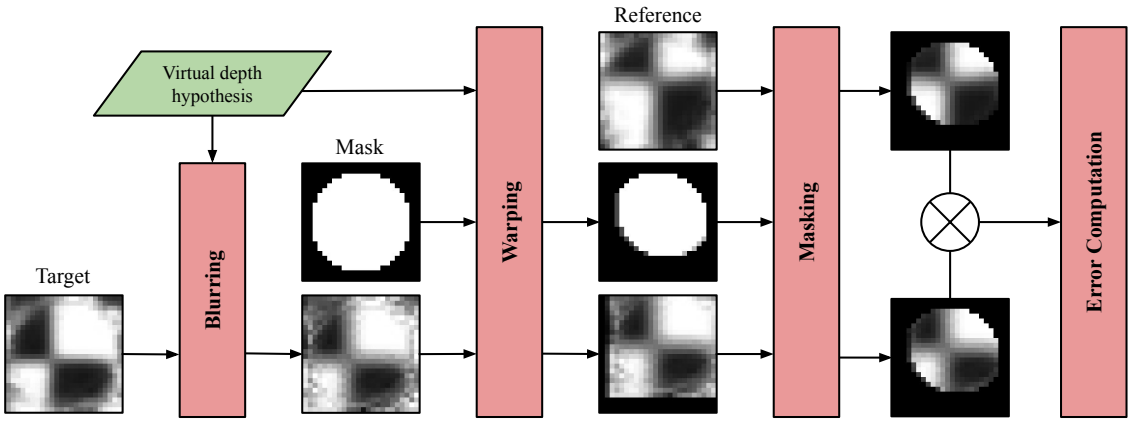


Figure 4.3: Overview of the proposed similarity error computation of our BLADE framework. First the target is equally-defocused given the virtual depth hypothesis. Second, the mask and the target are warped at the corresponding disparity. Third, the reference and the target are masked and compared to compute the similarity error.

4.3.1 Link between disparity and relative defocus blur

Recall that blur modeling has been addressed in subsubsection 3.1.3.3 and subsubsection 3.3.5.1. Blur can be linked to the depth information. We are able to characterize the blur profile of a plenoptic camera given the calibration parameters (see Figure 3.27 and Figure 4.4). We can make the following observations regarding the behaviors in the blur profiles depending on the considered range:

- First, as we want to compare blur between two micro-images, a projected point needs to be farther than the $\upsilon = 2$ limit to be observable into these two micro-images. Therefore, no relative blur estimation can be conducted for distances less than this limit (i.e., points at far distances in object space).

- The DoF, i.e., the region close to the focus plane where the blur size is below the pixel size, is a region where blur is not measurable.

- Far from the focus plane, the depth estimation accuracy is limited by the growth rate of the blur radius which tends to be constant (as the blur is linearly function of the inverse of the distance).

- Finally, the best estimations occur near the DoF where the micro-lenses are slightly out of focus, but the range of these distances is limited.

To overcome these drawbacks, we exploits the disparity information along with the blur information.

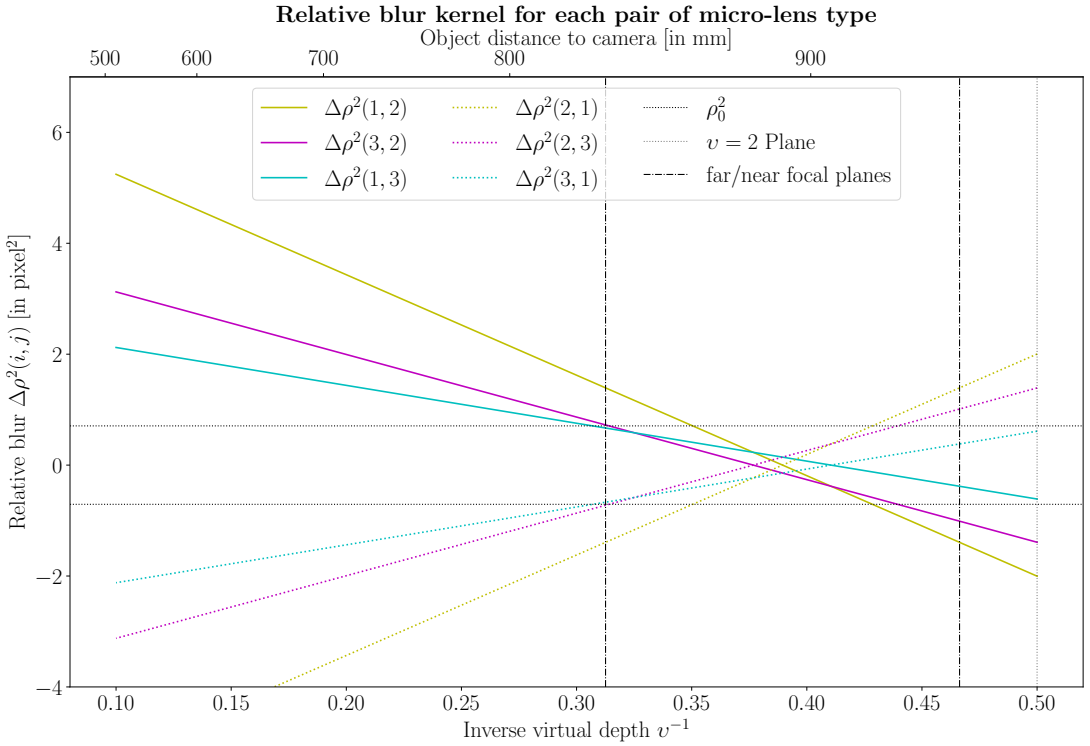


Figure 4.4: Relative blur as function of the inverse virtual depth, for each pair of micro-lens type, with a focus distance $h = 1000$ mm (from calibration of dataset `R12-B`). $\Delta\rho^2(i,j) > 0$ indicates that a pixel in the (j)-micro-image is more in-focus than its corresponding pixel in the (i)-micro-image, and vice-versa.

4.3.1.1 Defocus stereo images configuration

As seen previously, in standard stereo matching, two focused rectified stereo images are used to determine disparity. The left image $\mathcal{I}_{(i)}$ and right image $\mathcal{I}_{(j)}$ are related with the spatially-variant disparity $\boldsymbol{\delta} \in \mathbb{R}^2$ along the epipolar line, and thus the correspondence between the two images can be modeled as

$$\mathcal{I}_{(i)}(\boldsymbol{p}) = \mathcal{I}_{(j)}(\boldsymbol{p} + \boldsymbol{\delta}) \tag{4.5}$$

where $\boldsymbol{p} = (x, y)$ is the spatial index of a pixel. Taking into account blur, we model each image as the convolution of the in-focus image with a blur kernel as in Eq. (3.70) and we consider the equally defocused images as in Eq. (3.71). Therefore, injecting the disparity in the equally defocused model, the correspondence is given by

$$\begin{cases} \mathcal{I}_{(i)}(\boldsymbol{p}) \simeq \mathrm{h}_r * \mathcal{I}_{(j)}(\boldsymbol{p} + \boldsymbol{\delta}) & \text{if } \sigma_{(i)}(\boldsymbol{p}) \geq \sigma_{(j)}(\boldsymbol{p} + \boldsymbol{\delta}) \\ \mathrm{h}_r * \mathcal{I}_{(i)}(\boldsymbol{p}) \simeq \mathcal{I}_{(j)}(\boldsymbol{p} + \boldsymbol{\delta}) & \text{if } \sigma_{(i)}(\boldsymbol{p}) < \sigma_{(j)}(\boldsymbol{p} + \boldsymbol{\delta}) \end{cases}, \tag{4.6}$$

where h_r is the relative blur kernel of spread parameter σ_r applied to either one of the views such that both are equally-defocused.

4.3.1.2 Relative blur from images disparity using the S-Transform

From Subbarao and Surya [177], we can retrieve the focused image $\mathcal{I}^*$ using the S-Transform as

$$\mathcal{I}^*(\boldsymbol{p}) = \mathcal{I}(\boldsymbol{p}) - \frac{\sigma^2}{4} \cdot \nabla^2 \mathcal{I}(\boldsymbol{p}) \tag{4.7}$$

where σ is the spread parameter of the blur kernel, and ∇^2 is the Laplacian operator. Taking inspiration from the blur equalization technique (BET) [234] and using the equally defocused model with disparity (Eq. (4.6)), under the hypothesis that the (j)-view is more in-focus than the (i)-view (i.e., $\sigma_{(i)}(\boldsymbol{p}) \geq \sigma_{(j)}(\boldsymbol{p}+\boldsymbol{\delta})$), we derive

$$\begin{aligned} \mathcal{I}_{(i)}(\boldsymbol{p}) &\simeq \mathrm{h}_r * \mathcal{I}_{(j)}(\boldsymbol{p}+\boldsymbol{\delta}) \\ &= \mathcal{I}_{(j)}(\boldsymbol{p}+\boldsymbol{\delta}) + \frac{{\sigma_r}^2}{4} \cdot \nabla^2 \mathcal{I}_{(j)}(\boldsymbol{p}+\boldsymbol{\delta})\,. \end{aligned} \tag{4.8}$$

4.3.1.3 Relative blur as function of the virtual depth

On the other hand, from the previously derived blur radius formula (Eq. (3.39) and Eq. (3.64)), the relative blur can be approximated by a linear function of the disparity, i.e., of the inverse virtual depth up to a factor, such that

$$\Delta r^2 = r_{(i)}^2 - r_{(j)}^2 \approx m_{i,j} \cdot \upsilon^{-1} + q_{i,j} \quad [\mathrm{mm}^2] \tag{4.9}$$

with

$$m_{i,j} = \frac{\Delta_\mu^2 d}{2} \cdot \left(\frac{1}{{a_0}^{(i)}} - \frac{1}{{a_0}^{(j)}} \right) \tag{4.10}$$

and

$$q_{i,j} = \frac{\Delta_\mu^2 d^2}{4} \cdot \left(\left(\frac{1}{{a_0}^{(i)}} \right)^2 - \left(\frac{1}{{a_0}^{(j)}} \right)^2 \right), \tag{4.11}$$

where ${a_0}^{(i)}$ is the distance to the plane of focus of the type (i) micro-lens, computed as

$${a_0}^{(i)} = \frac{d f^{(i)}}{d - f^{(i)}}. \tag{4.12}$$

All the parameters are known thanks to the camera calibration. Finally, the spread parameter σ_r is computed using Eq. (4.9) and Eq. (3.23), such that

$$\sigma_r = \kappa \cdot \frac{1}{s} \cdot \left| \Delta r^2 \right|^{\frac{1}{2}}, \tag{4.13}$$

where κ is calibrated based on the procedure of section 3.3.5.

Note that the approximation is exact when considering that micro-lenses are parallel to the sensor plane. When dealing with micro-lenses in the same local neighborhood, the z-shift inducing a slight difference between the virtual depths can be neglected, and with orthogonal approximation, the relation stands (see Appendix D).

4.3.2 Blur aware depth estimation

We propose a process based on area matching techniques to estimate a raw depth map $\mathcal{D}$ directly from raw plenoptic images. Two variations are considered:

1. coarse estimation, i.e., one depth per micro-image, $\mathcal{D}(k,l)$;
2. refined estimation, i.e., one depth per pixel, $\mathcal{D}(x,y)$.

Figure 4.6 summarizes the estimation process in the coarse case. Figure 4.7 summarizes the estimation process in the refined case. Example of depth maps obtained by our method are illustrated in Figure 4.8. A new residual error is formulated to leverage blur information for depth estimation using a multi-focus plenoptic camera. The computation is illustrated in Figure 4.3. In the following, we detail each step of the process. Let an observation be a pair of micro-images such that the *reference* $\mathcal{I}^*$ is the most defocused and the *target* $\mathcal{I}$ is the micro-image to be equally defocused.

4.3.2.1 Disparity

Given a a virtual depth hypothesis v, the disparity $\delta = \|\boldsymbol{\delta}\|$ with $\boldsymbol{\delta} \in \mathbb{R}^2$ is obtained usually using the following relation

$$\boldsymbol{\delta} = \frac{1}{v} \cdot \boldsymbol{B} \quad \text{with} \quad \boldsymbol{B} = (\boldsymbol{C}^* - \boldsymbol{C}), \tag{4.14}$$

where $\boldsymbol{C}^*$ and $\boldsymbol{C}$ are respectively the centers of the reference and target micro-lenses in the MLA plane, and $\boldsymbol{B}$ is the baseline. This relation gives the disparity in case of orthogonal projection of micro-lens center to micro-image center. In other words, the relation would stand if the micro-images were extracted at $\boldsymbol{c}_0^*$ and $\boldsymbol{c}_0$. But in practice they are extracted at $\boldsymbol{c}^*$ and $\boldsymbol{c}$. To take into account the deviation of the micro-image centers (see Eq. (3.36)), the corrected disparity δ' in micro-image space is given by

$$\boldsymbol{\delta'} = \frac{(1-\lambda)\cdot v + \lambda}{v} \cdot \boldsymbol{B'} \tag{4.15}$$

with

$$\boldsymbol{B'} = \lambda \cdot \boldsymbol{B} = (\boldsymbol{c}^* - \boldsymbol{c}), \tag{4.16}$$

where $\lambda = D/(D+d)$ (see Eq. (3.38)), and $\boldsymbol{c}^*, \boldsymbol{c}$ are respectively the centers of the reference and target micro-images defining the baseline in image space such that $B' = \|\boldsymbol{c}^* - \boldsymbol{c}\|$.

4.3.2.2 Matching problem

Before triangulation, we need to know which pixels in neighbors micro-images are images of the same object point. This is the matching problem between views as in

standard stereo approaches. The correspondence is modeled by an affine warping function $\omega(\mathcal{I}, \boldsymbol{\delta})$ at the disparity hypothesis $\boldsymbol{\delta} \in \mathbb{R}^2$ along the epipolar line that is applied to rectify the image $\mathcal{I}$, such that

$$\omega(\mathcal{I}, \boldsymbol{\delta})\,(\boldsymbol{p}) = \mathcal{I}(\boldsymbol{p} + \boldsymbol{\delta})\,. \tag{4.17}$$

Pixel intensities are interpolated using bilinear-interpolation. Pixels which are not reprojected are set to 0. We can compare the warped target image with the reference image to estimate if the disparity hypothesis is correct. The similarity between the two image is calculated using the SAD measure (Eq. (4.1)).

4.3.2.3 Mask

To deal with circular micro-image of center $\boldsymbol{c}$ and radius ϱ, we define a mask image $\mathcal{M}$ by

$$\mathcal{M}(\boldsymbol{p}) = \begin{cases} 0 & \text{if } \|\boldsymbol{p} - \boldsymbol{c}\| > \varrho - b \\ 1 & \text{if } \|\boldsymbol{p} - \boldsymbol{c}\| \leq \varrho - b \end{cases} \tag{4.18}$$

where b is the margin border of the micro-image. In our experiments, we used $b = 1.5$ pixel to minimize the vignetting effect. The final mask $\mathcal{M}^*$ is given as the intersection of the circular mask and the warped circular mask, such that it represents the common pixels between the reference and the target at the given disparity hypothesis, i.e.,

$$\mathcal{M}^* = \mathcal{M} \circ \omega(\mathcal{M}, \boldsymbol{\delta})\,, \tag{4.19}$$

where $\circ$ is the element-wise matrix multiplication, i.e., the Hadamard matrix product. Examples of masked micro-images are given in Figure 4.3.

4.3.2.4 Blur equalization

One specificity of the multi-focus plenoptic camera is that for a same portion of a scene observed in micro-images using different focal lengths, these micro-images will demonstrate different amounts of blur. To compensate for the blur mismatch between the reference and the target, we use the equally defocused representation by adding supplemental amount of blur to the target image. The spread parameter σ_r is obtained from Eq. (4.13) at the given virtual depth hypothesis. To avoid dealing with micro-image borders while adding the relative blur, we use the S-Transform [177] from Eq. (4.7). The equally defocused target image $\bar{\mathcal{I}}$ is then computed as

$$\bar{\mathcal{I}} = \mathrm{h}_r * \mathcal{I} = \mathcal{I} + \frac{{\sigma_r}^2}{4} \cdot \nabla^2 \mathcal{I}, \tag{4.20}$$

where σ_r is the spread parameter of the blur kernel, and ∇^2 is the Laplacian operator.

4.3.2.5 Similarity error computation

The similarity residual error is computed as the normalized SAD between the masked reference image and the masked equally-defocused matched target image. It is expressed as

$$\begin{aligned}\varepsilon_{\text{sim}}(\mathcal{I}^*, \mathcal{I}, \boldsymbol{\delta}) &= \eta \cdot \sum_{\boldsymbol{p}} \left| \mathcal{I}^*(\boldsymbol{p}) - \omega\big(\bar{\mathcal{I}}, \boldsymbol{\delta}\big)(\boldsymbol{p}) \right| \cdot \mathcal{M}^*(\boldsymbol{p}) \\ &= \eta \cdot \left\| \big(\mathcal{I}^* - \omega\big(\bar{\mathcal{I}}, \boldsymbol{\delta}\big)\big) \circ \mathcal{M}^* \right\|_1 ,\end{aligned} \tag{4.21}$$

with η being the normalization factor defined as the sum of the common pixels between the target and the reference. This factor is given by

$$\eta^{-1} = \sum_{\boldsymbol{p}} \mathcal{M}^*(\boldsymbol{p}) = \|\mathcal{M}^*\|_1 . \tag{4.22}$$

Normalization of the error is required as the number of pixels to take into account varies with the disparity, and therefore with the virtual depth and the pair of micro-images. For a same virtual depth hypothesis, according to the pair of micro-images, the disparity will change.

4.3.2.6 Cost computation

For a micro-image $\mathcal{I}^*$, the cost is the weighted sum of all the errors computed for each micro-image $\mathcal{I}$ in its neighborhood $\mathcal{N}(\mathcal{I}^*, \upsilon)$ at the given virtual depth hypothesis υ. A plenoptic camera has a varying baseline for triangulation over the depth range, as a point is seen in more and more micro-images as $|\upsilon|$ increases, as illustrated in Figure 4.5. It means that $\mathcal{N}(\mathcal{I}^*, \upsilon)$ grows with υ, and must be retrieve at the correct hypothesis to take into account all observations. The total cost is then given by

$$\Theta(\mathcal{I}^*, \upsilon) = \frac{1}{W} \cdot \sum_{\mathcal{I} \in \mathcal{N}(\mathcal{I}^*, \upsilon)} \mathrm{w}(\mathcal{I}^*, \mathcal{I}) \cdot \varepsilon_{\text{sim}}(\mathcal{I}^*, \mathcal{I}, \boldsymbol{\delta}) , \tag{4.23}$$

with $\mathrm{w}(\mathcal{I}^*, \mathcal{I})$ being a weight function, and W being the total weight computed as

$$W = \sum_{\mathcal{I} \in \mathcal{N}(\mathcal{I}^*, \upsilon)} \mathrm{w}(\mathcal{I}^*, \mathcal{I}) . \tag{4.24}$$

In our experiment, we define the weight as constant.

4.3.2.7 Initialization

The number of MIs that see the same scene points in the neighborhood $\mathcal{N}(\mathcal{I}, \upsilon)$ of the considered MI depends on the virtual depth hypothesis. So, we first coarsely

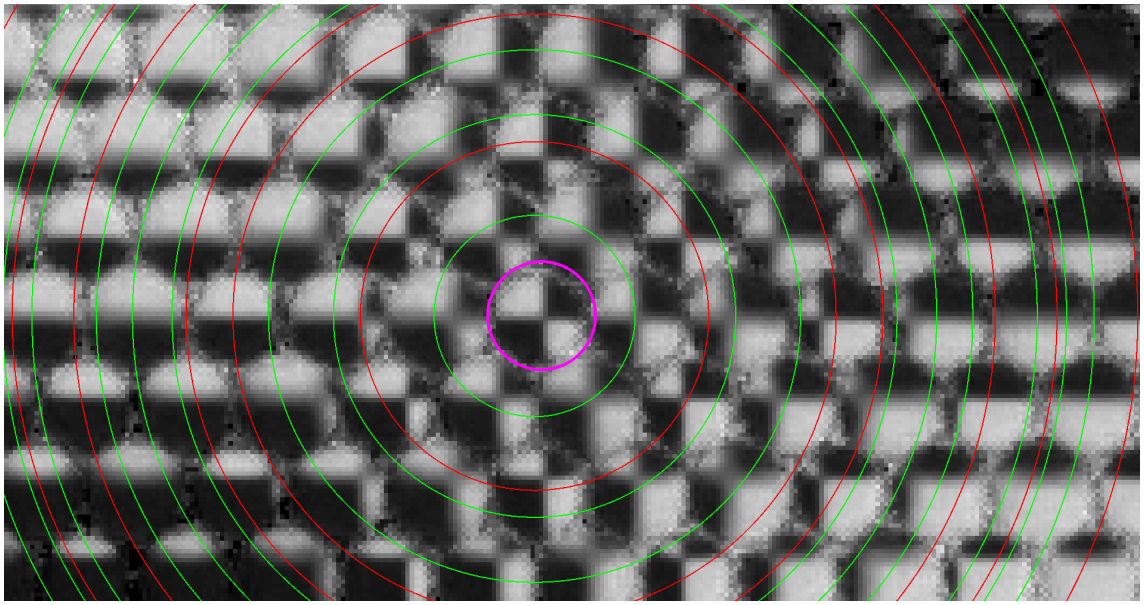


Figure 4.5: Graph of baselines representing a micro-image neighborhood. In *red*, the micro-images of the same type, and in *green* all the other baselines. The number of neighbors grows exponentially as the virtual depth, i.e., the maximum baseline, grows.

initialize v_0 from MIs of the same type (here, at baseline $B = 2 \cdot \sin \frac{\pi}{3}$, in case of a hexagonal MLA arrangement with three types of micro-lenses). We use then v_0 to retrieve the correct neighborhood size, and we restrict the search of the optimal value to $v_0 \pm N$, with $N = 1.96$ in our experiments.

4.3.2.8 Coarse depth estimation

Figure 4.6: Process of computing a coarse depth map $\mathcal{D}(k,l)$ using our blur aware depth estimation (BLADE) framework.

Input: Raw image, Camera model
Output: Coarse depth map $\mathcal{D}(k,l)$

1: **for all** micro-image $\mathcal{I}$ with enough texture **do** ▷ Eq. (4.26)
2: retrieve default neighborhood $\mathcal{N}(\mathcal{I})$
3: compute initial virtual depth v_0 ▷ Eq. (4.25)
4: $\mathcal{D}(k,l) \leftarrow v_0$
5: update neighborhood $\mathcal{N}(\mathcal{I}, v_0)$
6: compute virtual depth $\hat{v}$ ▷ Eq. (4.25)
7: $\mathcal{D}(k,l) \leftarrow \hat{v}$
8: **end for**
9: convert virtual to metric using $\Pi_{k,l}^{-1}$ ▷ Eq. (3.43)

Under the hypothesis of one depth per micro-image, i.e., corresponding to locally planar approximation, the coarse depth map $\mathcal{D}(k,l)$ is estimated as follows. For each micro-image, virtual depth estimation is conducted by a minimization of the latter cost function in an optimization process, such that

$$\hat{v} = \arg\min_{v} \Theta(\mathcal{I}^*, v)\,. \tag{4.25}$$

As the function is in 1-D, we use the golden search section (GSS) algorithm [235] to find the minimum with the desired precision. To improve time computation, only MIs with sufficient amount of texture are considered, such that

$$\mathrm{std}(\mathcal{I}, \mathcal{M}) > t_{\mathrm{c}}, \tag{4.26}$$

with $\mathrm{std}(\cdot,\cdot)$ being the standard deviation of the pixels intensity $\boldsymbol{p} \in \mathcal{I} \mid \mathcal{M}(\boldsymbol{p}) \neq 0$, and t_{c} being a threshold to reject non-textured area. In our experiments, we set $t_{\mathrm{c}} = 5$. An example of coarse virtual depth map is given in Figure 4.8(a).

4.3.2.9 Refined depth estimation

Figure 4.7: Process of computing a refined depth map $\mathcal{D}(x,y)$ using our blur aware depth estimation (BLADE) framework.

Input: Raw image, Camera model
Output: Refined depth map $\mathcal{D}(x,y)$

for all micro-image $\mathcal{I}$ **do**
  retrieve default neighborhood $\mathcal{N}(\mathcal{I})$
  for all pixel (x,y) with enough texture **in** $\mathcal{I}$ **do** ▷ Eq. (4.28)
    compute initial virtual depth v_0 ▷ Eq. (4.30)
    $\mathcal{D}(x,y) \leftarrow v_0$
  end for
  update neighborhood $\mathcal{N}(\mathcal{I}, v_0)$
  for all pixel (x,y) with enough texture **in** $\mathcal{I}$ **do** ▷ Eq. (4.28)
    compute virtual depth $\hat{v}$ ▷ Eq. (4.30)
    $\mathcal{D}(x,y) \leftarrow \hat{v}$
  end for
end for
convert virtual to metric using $\Pi_{k,l}^{-1}$ ▷ Eq. (3.43)

Under the hypothesis of one depth per pixel, a refined depth map $\mathcal{D}(x,y)$ is computed. The virtual depth estimation is conducted in a similar fashion as for the coarse estimation. For a pixel $\boldsymbol{p} = (x,y)$, errors and costs are computed the same way as previously but considering only the result within a window $\mathcal{W}$ extracted

around $\boldsymbol{p}$. In our experiments, we use a window of size 5×5 pixel. The similarity residual error $\varepsilon_{\text{sim}}(\mathcal{I}^*, \mathcal{I}, \boldsymbol{\delta}, \boldsymbol{p})$ is then given by

$$\|\mathcal{W}(\boldsymbol{p}, \mathcal{M}^*)\|_1^{-1} \cdot \left\|\mathcal{W}\big(\boldsymbol{p}, \big(\mathcal{I}^* - \omega\big(\bar{\mathcal{I}}, \boldsymbol{\delta}\big)\big) \circ \mathcal{M}^*\big)\right\|_1 . \tag{4.27}$$

The cost at a pixel $\boldsymbol{p}$ having a sufficient contrast, i.e., such that it verifies

$$\text{std}(\mathcal{W}(\boldsymbol{p}, \mathcal{I}), \mathcal{W}(\boldsymbol{p}, \mathcal{M})) > t_{\text{c}}, \tag{4.28}$$

is given by

$$\Theta(\mathcal{I}^*, \upsilon, \boldsymbol{p}) = \frac{1}{W} \cdot \sum_{\mathcal{I} \in \mathcal{N}(\mathcal{I}^*, \upsilon)} \text{w}(\mathcal{I}^*, \mathcal{I}) \cdot \varepsilon_{\text{sim}}(\mathcal{I}^*, \mathcal{I}, \boldsymbol{\delta}, \boldsymbol{p}) , \tag{4.29}$$

where the weights are defined as previously (see Eq. (4.24)). Finally, we compute the virtual depth $\hat{\upsilon}$ at each pixel $\boldsymbol{p}$ as follows

$$\hat{\upsilon} = \arg\min_{\upsilon} \Theta(\mathcal{I}^*, \upsilon, \boldsymbol{p}) . \tag{4.30}$$

An example of refined virtual depth map is given in Figure 4.8(b).

4.3.3 Depth scaling calibration

At the end of the depth estimation process, we have a virtual depth estimate associated to each pixel. To obtain a metric depth information, we can use the inverse projection model given in Eq. (3.43), where ρ is computed from the virtual depth by Eq. (3.64).

First, we make the observation that the algorithm effectively retrieves the virtual depth hypothesis corresponding to the observed images, i.e., the relation

$$\hat{\upsilon} = \frac{B}{B - \Delta\boldsymbol{p}} \tag{4.31}$$

is verified, where $\Delta\boldsymbol{p}$ is the Euclidean distance between two observations. However, when mapping from the virtual space to the object space with the inverse projection model, we observe a significant error in the z-dimension (e.g., points are projected farther) but also in the xy-dimension (e.g., objects appear bigger). Reconstructed objects are scaled up to a certain factor, which grows approximately linearly as function of the distance with respect to the focus plane (see Figure 4.11). This phenomenon appears both on real and simulated data.

It is due to the limitations of the thin-lens model. The proposed model describes efficiently the projective geometry for a point whose chief ray attains the desired pixel, but this is not always the case. Given specific setup configuration and with

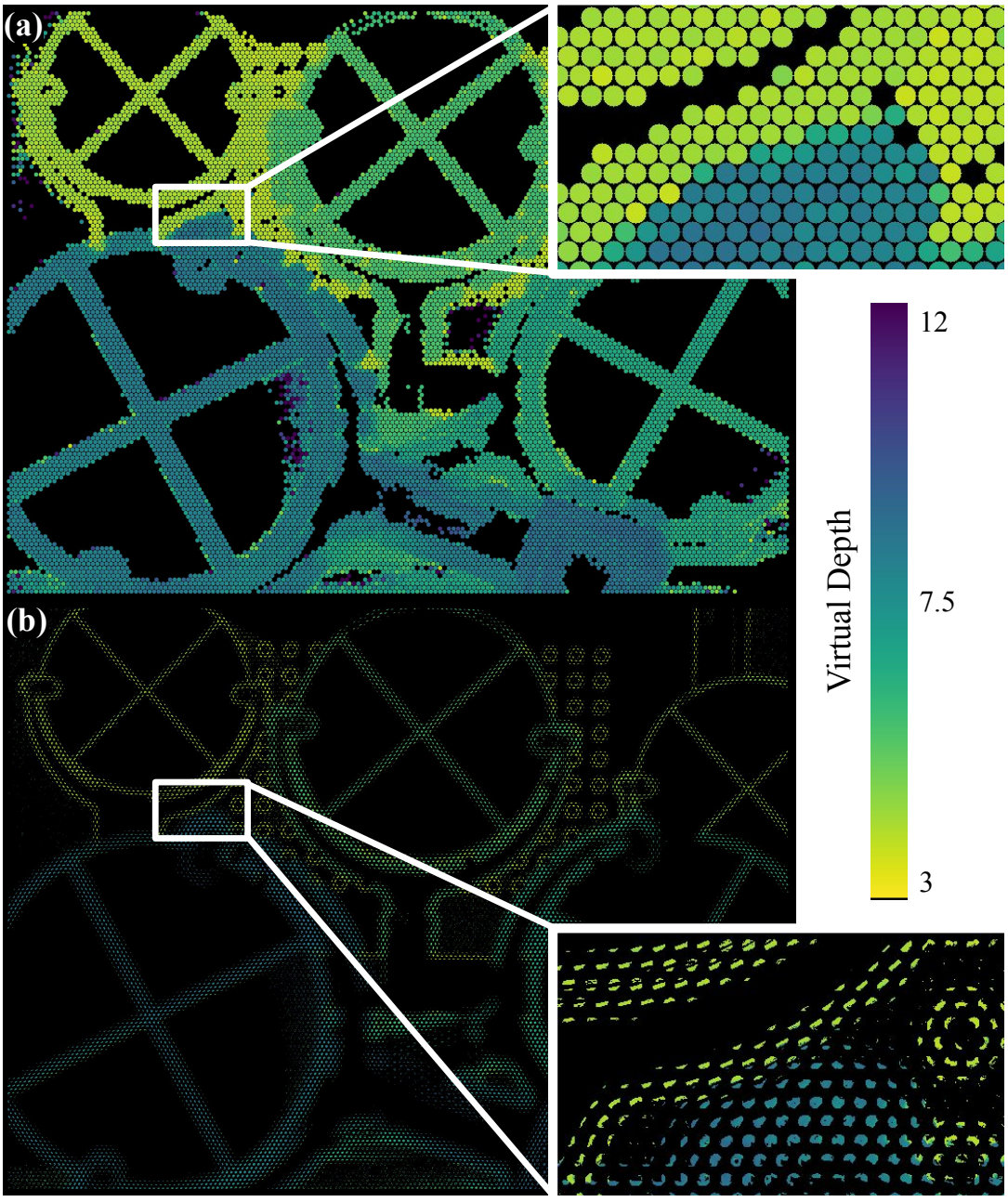


Figure 4.8: Examples of raw virtual depth maps obtained by our BLADE framework with a zoom on an occluded area. **(a)** Coarse virtual depth map $\mathcal{D}(k,l)$, with one estimation per micro-image. **(b)** Refined virtual depth map $\mathcal{D}(x,y)$, with one estimation per pixel. More details can be captured per micro-image, but the map is sparser.

aperture corresponding to the f-number matching principle, not all rays from the cone of light reach the pixel, inducing a shift in the radiance.

Although not explicitly pointing out this issue, Zeller *et al.* [230] calibrated the mapping between virtual depth and metric depth in object space, by proposing three different models. Heinze *et al.* [125] also noted the presence of a systematic error in the depth estimation process, even after correcting the offset induced by the thick-lens model. In our case, we are able to measure and characterize this scaling error, and thus we can correct it in a post-calibration process.

4.3.3.1 Scaling error measurement

To quantify the scale error, we use the relative mean bias error (MBE) to measure the relative difference of the distance between each pair of back-projected corners $\|\bar{\boldsymbol{p}}_i - \bar{\boldsymbol{p}}_j\|$ and the known distance $\|\boldsymbol{p}_i - \boldsymbol{p}_j\|$ between these corners of a checkerboard. Scaling error measurement is done in four-steps:

1. We perform depth estimation with our BLADE framework on plenoptic raw image of a checkerboard.
2. We back-project each BAP feature $\boldsymbol{p}_{k,l}$ having a virtual depth v of the same cluster $\mathcal{C}_i$, i.e., corresponding to the same corner $\boldsymbol{p}_i$, using Eq. (3.43).
3. We compute then the centroid $\bar{\boldsymbol{p}}_i$ corresponding to the checkerboard corner $\boldsymbol{p}_i$, as
$$\bar{\boldsymbol{p}}_i = \frac{1}{\#\mathcal{C}_i} \cdot \sum_{\boldsymbol{p}_{k,l} \in \mathcal{C}_i} \Pi_{\mathrm{k,l}}^{-1}(\boldsymbol{p}_{k,l})\,, \tag{4.32}$$
where $\#\mathcal{C}_i$ is the number of observations in the cluster $\mathcal{C}_i$.
4. The scale error $\varepsilon_{\mathrm{scale}}$, for a frame having $I \times J$ corners, is finally computed as
$$\varepsilon_{\mathrm{scale}} = \frac{1}{I \cdot J} \cdot \sum_{(i,j) \in I \times J} \left(1 - \frac{\|\bar{\boldsymbol{p}}_i - \bar{\boldsymbol{p}}_j\|}{\|\boldsymbol{p}_i - \boldsymbol{p}_j\|}\right). \tag{4.33}$$

4.3.3.2 Scale correction model

From the observed data (see Figure 4.11), we can infer that the scaling error is function of the distance z, and that a linear or a quadratic function is sufficient to fit the results. We proposed then to model the scaling correction either as a linear or as a quadratic function, noted $\Gamma(\cdot)$. The function Γ takes as input the z-component of the 3D point obtained by back-projection, and the point is then re-scaled by
$$\gamma = \frac{\Gamma(z)}{z}, \tag{4.34}$$

i.e., the corrected point $\boldsymbol{p}^*$ from a pixel $\boldsymbol{p}_{k,l}$ having a virtual depth υ is

$$\boldsymbol{p}^* = \gamma \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \text{ where } \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \propto \Pi_{\mathrm{k,l}}^{-1} \left(\begin{bmatrix} u \\ v \\ \rho = m \cdot \upsilon^{-1} + q_i \end{bmatrix} \right). \tag{4.35}$$

Both linear and quadratic models fit well the data, and allow to correct the scale error as shown by Figure 4.11. The quadratic model is better at correcting depth all across the range of depths, and the final correction have a lower error and is nearly constant for all depths.

4.3.3.3 Depth scaling calibration

At this point, the camera intrinsic parameters, the relative blur coefficient and the inverse distortion coefficients have been calibrated. Depth estimation based on the BLADE framework can therefore be applied on raw plenoptic images. We propose then a post-calibration process for the scaling correction based on non-linear optimization of the scale error over several checkerboard raw plenoptic images. Let $\Xi = \{\gamma_0, \gamma_1, \gamma_2\}$ be the set of parameters to optimize, such that

$$\Gamma(z) = \gamma_2 z^2 + \gamma_1 z + \gamma_0. \tag{4.36}$$

The cost function $\Theta(\Xi)$ is expressed as the sum over each frame n of the scale errors $\varepsilon_{\text{scale}}$, i.e.,

$$\Theta(\Xi) = \frac{1}{IJN} \cdot \sum_n \sum_{(i,j) \in I \times J} \left(1 - \frac{\left\| \gamma_i \cdot \bar{\boldsymbol{p}}_i^n - \gamma_j \cdot \bar{\boldsymbol{p}}_j^n \right\|}{\left\| \boldsymbol{p}_i^n - \boldsymbol{p}_j^n \right\|} \right), \tag{4.37}$$

where N is the number of frames and $I \cdot J$ is the number of checkerboard corners. Each point $\bar{\boldsymbol{p}}_i^n$ is re-scaled by γ_i as defined in Eq. (4.34) and Eq. (4.35). The optimization is conducted using the Levenberg-Marquardt algorithm.

4.4 Experimental validation

In this section we present the validation of our blur aware depth estimation framework. First, we analyze the corrected disparity and the scaling error modelings. Second, we compare our method on relative depth estimation with state-of-the-art methods, including the `Raytrix` software, corresponding to the model of Heinze *et al.* [125], and estimation with the model of Noury *et al.* [166] using only the disparity. Finally, we evaluate the depth estimation on real-world 3D complex scenes with ground truths acquired with a lidar. Our experimental setup is illustrated in Figure 4.9.

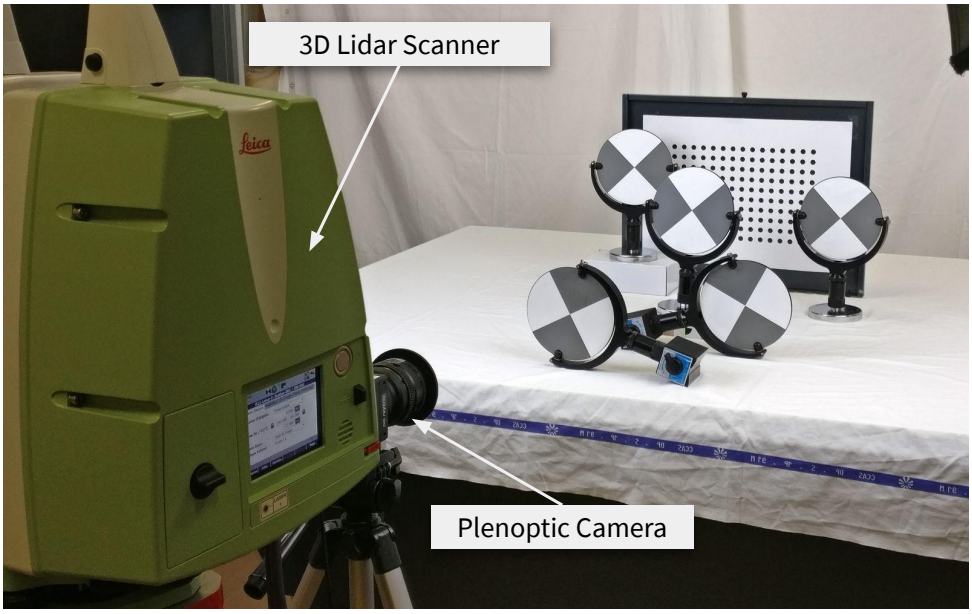


Figure 4.9: Our `Raytrix R12` multi-focus plenoptic camera in our experimental setup, capturing a scene for 3D reconstruction. A `Leica ScanStation P20` allows us to capture a colored point cloud that can be used as ground truth data.

4.4.1 Experimental setup

4.4.1.1 Hardware environment

Camera setup. For our experiments we used a `Raytrix R12` color 3D-light-field-camera, with a MLA of F/2.4 aperture. The camera is in Galilean *internal* configuration, i.e., the micro-lens focal lengths are greater than the distance MLA-sensor. The mounted lens is a `Nikon AF Nikkor F/1.8D` with a 50 mm focal length. The MLA organization is hexagonal row-aligned, and composed of 176×152 (width $\times$ height) micro-lenses with $I = 3$ different types. The sensor is a `Basler beA4000-62KC` with a pixel size of $s = 0.0055$ mm. The raw image resolution is 4080×3068 pixel. We used four focus distance configurations, with $h \in \{450, 1000, 2133, \infty\}$ mm. Note that when changing the focus setting, the main lens moves with respect to the block MLA-sensor.

Relative depth setup. The camera is mounted on a linear motion table with micro-metric precision. The target plane is orthogonal to the translation axis, and the camera optical axis is aligned with this axis. Images with known relative translation between each frame are then used to estimate depths and compared to the ground truth. This setup corresponds to the evaluation experimental setup presented in section 3.4.1. Relative errors are computed as presented in Section 3.4.2.3 by Eq. (3.77).

3D scenes setup. We used a 3D lidar scanner, a `Leica ScanStation P20 (LP20)`, that allowed us to capture a color point cloud with high precision that is used as ground truth data. The `LP20` was configured with no high dynamic range (HDR) and with a resolution of 1.6 mm @ 10 m.

4.4.1.2 Software environment

All images have been acquired using the `MultiCamStudio` free software (v6.15.1.3573) of the `Euresys` company. We set the shutter speed to 5 ms. For `Raytrix` data, we use their proprietary software `RxLive` (v4.0.50.2) to calibrate the camera, and compute the depth maps used in the evaluation.

4.4.1.3 Datasets

For the relative depth evaluation, we used the datasets presented previously. We evaluated then our depth estimation framework for focus distances $h \in \{450, 1000, \infty\}$ mm, corresponding to datasets `R12-A,B,C` respectively.

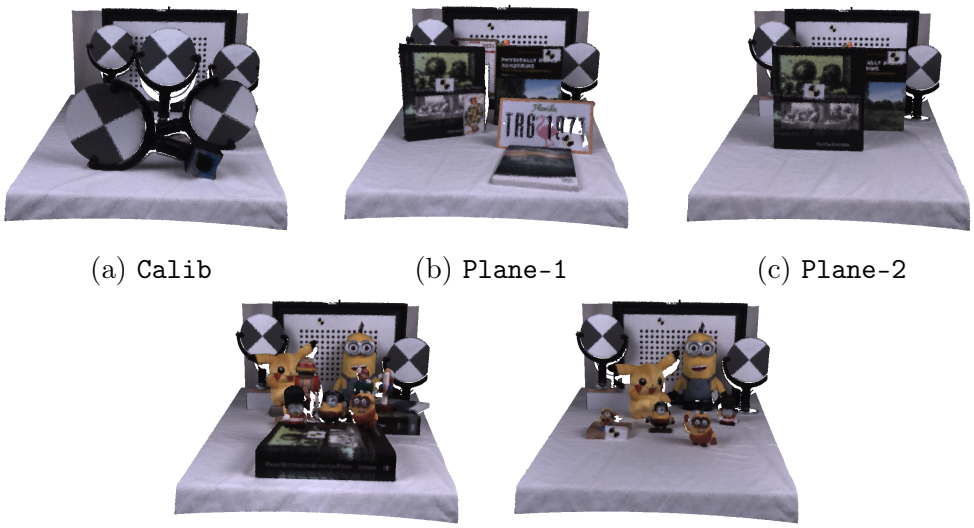
For the 3D scenes evaluation, we introduced a new dataset, namely `R12-E`, corresponding to a focus distance $h = 2133$ mm. The camera has been calibrated using our methodology presented in section 3.3.4. From this configuration, we created two sub-datasets:

1. a simulated dataset built upon our own simulator based on raytracing to generate images with known absolute position, named `R12-ES`;
2. a dataset composed of several 3D scenes with ground truth acquired with the `LP20`, for object distances ranging from 400 mm to 1500 mm.

The latter dataset, named `R12-ELP20`, includes fives scenes:

- one scene for extrinsic parameters calibration, containing checker corner targets, named `Calib`;
- two scenes containing textured planar objects, named `Plane-1` and `Plane-2`;
- and two more complex scenes containing various figurines, named `Figurines-1` and `Figurines-2`.

Each scene is composed of: a colored point cloud (with spatial (x, y, z) information, color information (r, g, b), and intensity information) in format `.ptx`, `.pts` and `.xyz`; 3D positions of the targets in the lidar reference frame; two raw plenoptic images in `rgb` color and two raw plenoptic images in bayer; and, photos and labels of the scene. Scene snapshot views of the dataset are given in Figure 4.10.



(a) `Calib` (b) `Plane-1` (c) `Plane-2`

(d) `Figurines-1` (e) `Figurines-2`

Figure 4.10: Scene snapshot views of dataset `R12-ELP20`.

4.4.2 Lidar-camera calibration

In order to transform the point cloud data $\boldsymbol{P}_l$ in the same reference frame as the camera, we calibrate the extrinsic parameters between those frames, i.e., the transformation ${}^c\boldsymbol{T}_l \in \mathrm{SE}(3)$ such as $\boldsymbol{P}_c = {}^c\boldsymbol{T}_l \boldsymbol{P}_l$, where $\boldsymbol{P}_c$ is the point cloud data expressed in the camera frame. The calibration is a four-steps process:

1. We acquired a point cloud of a scene containing calibration targets, and associated the 3D coordinates $\boldsymbol{p}_l$ of the corners manually from the point cloud. This set of corners forms a points constellation, noted $\mathcal{C}_l$.

2. A raw plenoptic image of the same scene is acquired with the plenoptic camera, and BAP features $\boldsymbol{p}_{k,l}$ are extracted. The features are clustered and associated to the points constellation. For each cluster of observations, the barycenter is computed. Those barycenters can be seen as the projections of the points constellation through the main lens using a standard pinhole model.

3. The initial transformation ${}^c\hat{\boldsymbol{T}}_l$ is thus estimated using the PnP algorithm [179], like in classic pinhole imaging system.

4. The transformation ${}^c\boldsymbol{T}_l$ is refined by minimizing the reprojection error of the points constellation, such that

$$\underset{{}^c\boldsymbol{T}_l}{\arg\min} \sum_{\boldsymbol{p}_l \in \mathcal{C}_l} \sum_{k,l} \left\| \boldsymbol{p}_{k,l} - \Pi_{k,l} \left({}^c\boldsymbol{T}_l \boldsymbol{p}_l \right) \right\|^2 . \tag{4.38}$$

The point cloud data $\boldsymbol{P}_l$ can now be expressed in the camera frame. It is thus used as ground truth for quantitative evaluations.

Table 4.1: Depth scaling coefficients for datasets `R12-A,B,C` and `R12-E,ES` with the median scale error after correction. For dataset `R12-C`, the error for the linear model is also reported.

	$\gamma_2\,(\times 10^{-5})$	γ_1	γ_0	$\varepsilon_{\text{scale}}\,(\%)$
`R12-A`	79.709	0.625	31.512	−0.014
`R12-B`	23.017	0.796	10.913	0.022
`R12-C`	2.736	0.883	10.910	0.024
`R12-C` (linear)	-	0.929	−6.101	0.043
`R12-ES`	14.431	0.667	35.910	−0.023
`R12-E`	4.617	0.912	−2.004	0.047

4.4.3 Depth scaling calibration results

4.4.3.1 Corrected disparity

We first evaluated the impact of the orthogonal approximation of the micro-lens baseline with respect to the micro-image baseline, i.e., using the corrected disparity (corresponding to Eq. (4.15)) instead of the commonly used disparity formulation (corresponding to Eq. (4.14)). Analysis is performed on dataset `R12-A`. Without depth scaling correction, we have for the approximated disparity a mean relative error $\varepsilon_z = 19.38$ %, which is reduced to $\varepsilon_z = 14.99$ % only by using the corrected disparity formulation. In the following, we will use the disparity obtained from Eq. (4.15) for all evaluations.

4.4.3.2 Depth scaling correction

Coarse depth estimation for dataset `R12-A,B,C,ES` is performed on images corresponding to planar checkerboards orthogonal to the optical axis and uniformly distributed in the range of distances, whilst for `R12-E`, depth estimation is performed on free-hand checkerboards, leading to noisier depth estimates. The depth associated to each frame is the median of the depth estimates. Calibrated depth scaling coefficients are reported in Table 4.1, along with their median scale error after correction for the evaluation datasets. Depths before and after correction are illustrated in Figure 4.11. A positive error means the estimated object is smaller than the ground truth, and a negative error means that the estimated object is bigger than the ground truth. The absolute error grows when getting farther from the focus distance. All corrected distances have a nearly null scale error. For all datasets, our methodology successfully corrects the scale, with a final median scale error $\varepsilon_{\text{scale}}$ of less than 0.05 %.

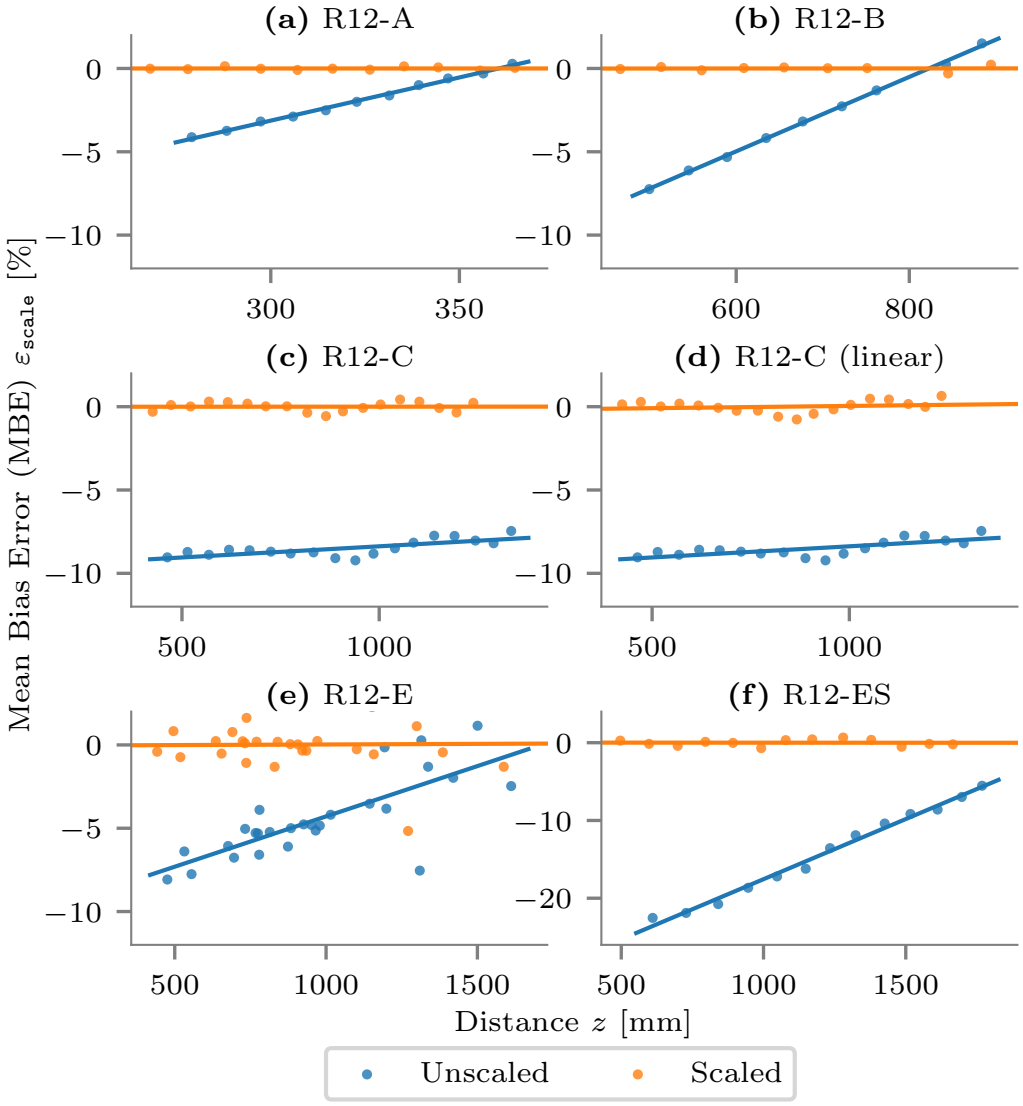


Figure 4.11: Scale errors before and after correction as function of the distance for the datasets `R12-A` **(a)**, `R12-B` **(b)**, `R12-C` **(c)**, `R12-E` **(e)** and `R12-ES` **(f)** with their associated fitting functions. A positive error means the estimated object is smaller than the ground truth, and a negative error means that the estimated object is bigger than the ground truth. The absolute error grows when getting farther from the focus distance. The corrected distance has a nearly null error. Sub-figures **(c)** and **(d)** give a comparison of the linear model versus the quadratic model scale correction on the dataset `R12-C`. The quadratic model performs slightly better than the linear model, as the fitted line slope is closer to zero compared to the slope of the linear fitted model.

4.4.3.3 On simulated data

We investigated scale error on simulated images of the `R12-ES` dataset. Depths are estimated based on the coarse depth estimation framework, for ground truth distances from 500 mm to 1900 mm with a step of 100 mm. Without scale correction, depths are estimated from 610.9 mm to 1917.4 mm with a mean relative error $\varepsilon_z = 8.06$ %. After scale correction, depths are estimated from 495.7 mm to 1841.8 mm with a mean relative error reduced by a factor two, $\varepsilon_z = 3.78$ %. As shown in Figure 4.11, depth scaling error appears even in simulated data, i.e., due to the finite aperture not allowing all rays to go through, showing that this phenomenon must be added to the inverse projection model to reach precise and accurate depth measurements.

4.4.3.4 Linear vs quadratic

Secondly, we evaluated the choice of the scaling model and presented the results for the dataset `R12-C`. As illustrated in Figure 4.11, The quadratic model performs slightly better than the linear model, as the fitted line slope is closer to zero compared to the slope of the linear fitted model. This is confirmed by the median scale error after correction reported in Table 4.1 which is reduced by a factor two with the quadratic model. In the following, depths will be corrected with the quadratic model.

4.4.4 Relative depth estimation results

We compared our method with (`BLADE`) and without (`BLADEu`) depth scale correction on relative depth estimation with state-of-the-art methods, including the `Raytrix` software, corresponding to the model of Heinze *et al.* [125] (`RTRX`), and depth estimation with the model of Noury *et al.* [166] using only the disparity (`DISP`). The corresponding intrinsic parameters are recalled in Table 4.3. The relative depth errors along the z-axis with respect to the ground truth displacement from the closest frame are reported in Figure 4.12 for datasets `R12-A` (a), `R12-B` (b) and `R12-C` (c). The mean error with its confidence interval across all datasets is illustrated in (d) for each method, and is:

- for `BLADE`, $\varepsilon_z = 4.09 \pm 0.85$ %;
- for `BLADEu`, $\varepsilon_z = 11.05 \pm 6.61$ %;
- for `DISP`, $\varepsilon_z = 7.88 \pm 3.74$ %;
- for `RTRX`, $\varepsilon_z = 6.76 \pm 3.96$ %.

First, we see that our scale correction effectively improves the depth estimation results. With scale correction it outperforms the other methods. This behavior is also validated in simulation on `R12-ES`, where the mean error after correction is of the same order, i.e., $\varepsilon_z = 3.78$ %.

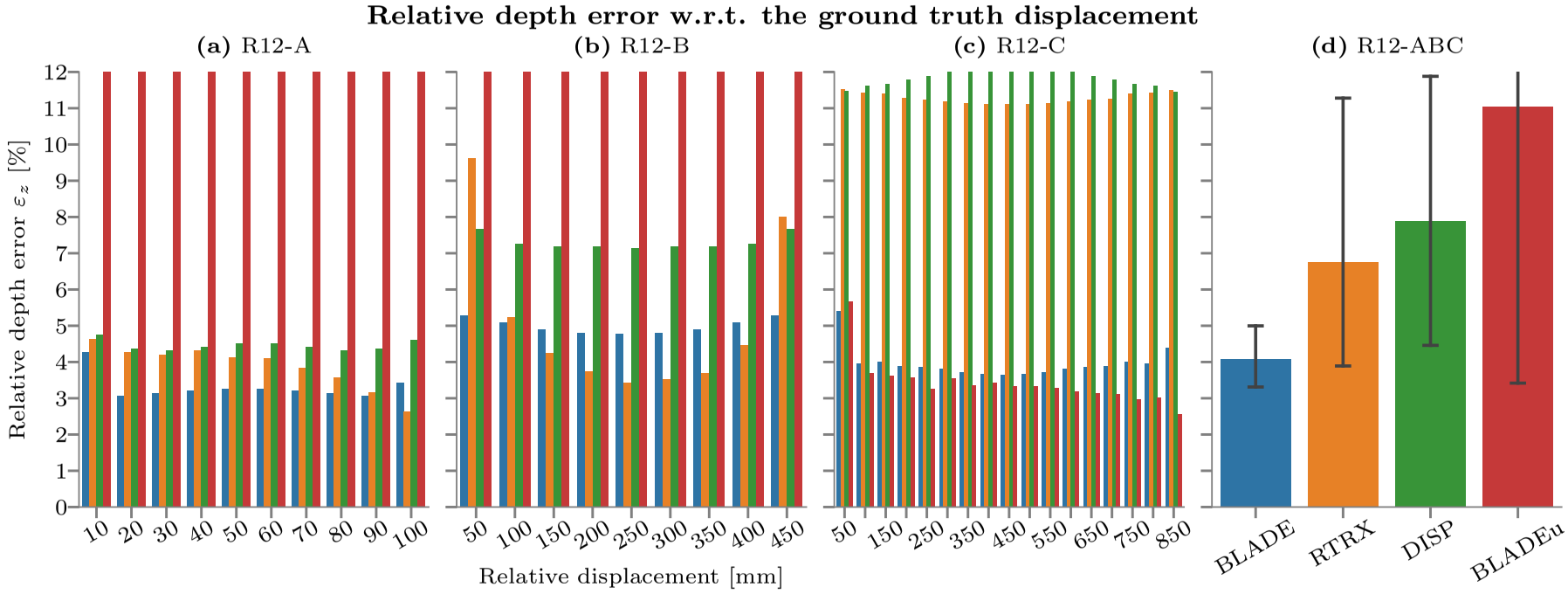


Figure 4.12: Relative depth error along the z-axis with respect to the ground truth displacement from the closest frame, for datasets `R12-A` **(a)**, `R12-B` **(b)** and `R12-C` **(c)**. The error ε_z is expressed in percentage of the estimated distance, and truncated to 12 % to ease the readability and the comparison. The mean error with its confidence interval across all datasets for our method with (`BLADE`) and without scale correction (`BLADEu`), for the model of [166] using only disparity (`DISP`), and for the proprietary software `RxLive` (`RTRX`) are reported in **(d)**. Please refer to the color version for better visualization.

Secondly, the `BLADE` method presents the lowest standard deviation, a stable error across all distances, and errors of the same order for all configurations. The other methods vary significantly across the datasets, making our method the only one able to generalize to several configurations without losing precision.

Finally, for datasets `R12-A,B,ES`, the scale correction clearly improves the relative depth estimates. This is not the case for `R12-C`, where the results are similar with and without correction. One explanation is that as the camera is focused at infinity, the working range of distances does not describe efficiently the scale error, which is nearly constant when uncorrected as illustrated in Figure 4.11, whereas the intrinsic parameters have been optimized for this range. With scale correction, the estimates range from 417.22 mm to 1229.83 mm, whereas the uncorrected depths range from 453.57 mm to 1325.43 mm. As we do not have absolute ground truth for the depth estimates, we cannot draw conclusion on which depth range is better than the other. So we proposes a new setup with absolute ground truth to evaluate the scale correction on 3D scenes.

4.4.5 Absolute depth evaluation on 3D scenes results

We used the dataset `R12-ELP20` to evaluate our depth estimation framework on absolute metric depth estimates.

4.4.5.1 Central sub-aperture depth map (CSAD)

To compare depth estimates, we generated for each scene the central sub-aperture depth map (CSAD) using the following methodology:

1. From the raw depth map, we back-project each pixel having a virtual depth hypothesis into a 3D point in metric space;
2. We replace the plenoptic camera model by a pinhole model, where the sensor is now at a distance F from the main lens, and we increase the pixel size by a factor S (here, $S = 4$, the final resolution is thus 1020×767 pixels);
3. We project each point of the point cloud with the new pinhole model, using a z-buffer like technique, and attributing the minimum depth value to the pixel (or the median value for noisy data);
4. And finally, we filter the resulting depth map image by applying a median filter and a morphological erosion to reduce noise.

In order to generate ground truth CSADs, we replace the first step by simply applying the extrinsic transformation ${}^{c}\boldsymbol{T}_{l}$ so that the 3D lidar points are expressed in camera

frame. In our experiments, using a constellation of five points, the obtained optimized transformation is

$$ {}^{c}\boldsymbol{T}_{l} = \begin{bmatrix} -0.88925 & -0.070239 & 0.451992 & 177.278 \\ -0.456949 & 0.0917283 & -0.88475 & -394.36 \\ 0.020684 & -0.993304 & -0.11366 & -266.858 \\ 0 & 0 & 0 & 1 \end{bmatrix}. $$

4.4.5.2 Evaluated methods

We evaluated our `BLADE` framework considering the following variations:

- using relative blur information (`B`) or only disparity (`D`);
- using the coarse (`C`) or the refined (`R`) estimation;
- and using the scale-corrected (`S`) or scale-uncorrected (`U`) model.

Note that we use the same intrinsic parameters for all evaluations. In the end, we presented the results for eight methods. For each method, we generated CSADs for each of the five scenes of `R12-ELP20`, and compared them to the ground truths.

4.4.5.3 Metrics

To analyze the depth error, we computed a quality map as the absolute difference (AD) between the depth map and the ground truth. As the maps are sparse, we compute a mask corresponding to pixels in common where depth estimates are available. Errors are computed only for pixels in the mask. Statistics over the errors are then computed. We used the percentiles (at 25 and 75 %) and the median to describe the overall error of the depth map estimates.

4.4.5.4 Results

Statistics for all the variations of the `BLADE` framework for all the scenes are reported in Table 4.2. Bold font indicates the best results. The last columns indicate the mean of the median errors for all scenes. Snapshot of the colored point cloud, along with the ground truth CSAD are reported for each scene in Figure 4.13 and Figure 4.14. Depth map, mask and quality map are illustrated for all variations using the relative blur (`B`) in Figure 4.13, and for all variations using only the disparity (`D`) in Figure 4.14.

From the reported errors, the errors distributions are similar for all the scenes. The scenes can be divided into two groups:

Table 4.2: Statistics (percentiles and median) of the absolute difference (AD) error of the central sub-aperture depth map (CSAD) for each variation of our BLADE framework (using relative blur information (B) or only disparity (D); using the coarse (C) or the refined (R) estimation; and using the scale-corrected (S) or scale-uncorrected (U) model), on the scenes of dataset R12-ELP20. The first table reports the easy scenes results, with mean of the median errors as last column. The second table reports the complex scenes results, with mean of the median errors as last column. The last table indicates the mean of the median errors for all scenes. All errors are expressed in mm.

B/D	C/R	S/U	Calib			Plane-1			Plane-2			Total
			$\mathcal{Q}_{25}$	med.	$\mathcal{Q}_{75}$	$\mathcal{Q}_{25}$	med.	$\mathcal{Q}_{75}$	$\mathcal{Q}_{25}$	med.	$\mathcal{Q}_{75}$	
B	C	S	**7.661**	**17.761**	**39.146**	**6.498**	**13.940**	**26.402**	**5.482**	**11.370**	**24.468**	**14.357**
D	C	S	7.916	18.648	39.156	7.002	14.672	27.599	5.644	12.007	25.022	15.109
B	R	S	8.831	19.961	43.880	11.765	21.818	34.903	9.758	17.951	28.925	19.910
D	R	S	9.736	21.992	49.872	14.800	25.904	39.295	11.074	20.613	32.054	22.836
B	C	U	34.450	50.106	68.997	30.743	42.452	58.241	33.657	42.905	55.664	45.154
D	C	U	31.105	47.071	66.983	27.581	39.358	54.387	30.754	40.081	51.925	42.170
B	R	U	27.114	41.327	63.583	17.553	28.304	40.732	20.025	30.192	39.835	33.274
D	R	U	26.274	41.522	64.952	14.098	24.255	37.062	17.115	27.317	38.432	31.031

B/D	C/R	S/U	Figurines-1			Figurines-2			Total
			$\mathcal{Q}_{25}$	med.	$\mathcal{Q}_{75}$	$\mathcal{Q}_{25}$	med.	$\mathcal{Q}_{75}$	
B	C	S	**11.720**	**24.867**	**47.270**	**12.310**	27.058	54.395	**25.963**
D	C	S	12.446	26.307	50.554	13.514	29.809	57.666	28.058
B	R	S	12.011	25.282	50.377	14.203	29.442	56.900	27.362
D	R	S	13.551	28.016	55.308	16.911	34.125	62.574	31.071
B	C	U	21.417	43.577	67.571	18.415	37.834	63.522	40.706
D	C	U	18.325	40.323	63.975	15.713	33.372	58.125	36.847
B	R	U	16.539	30.243	52.072	14.322	28.878	48.591	29.560
D	R	U	15.272	28.690	50.152	12.787	**26.733**	**46.932**	27.711

B/D	C/R	S/U	Total
B	C	S	**18.999**
D	C	S	20.289
B	R	S	22.891
D	R	S	26.130
B	C	U	43.375
D	C	U	40.041
B	R	U	31.789
D	R	U	29.703

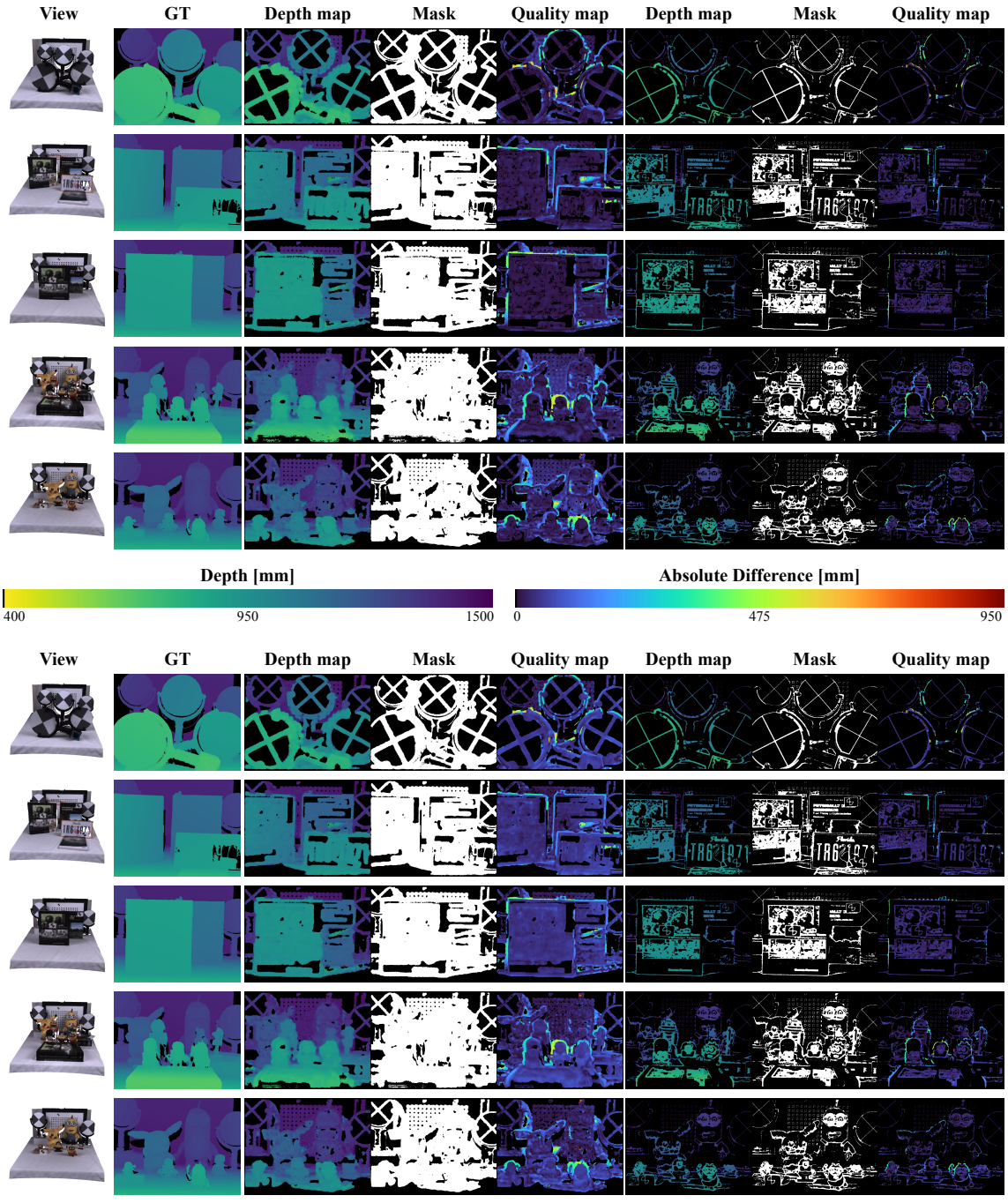


Figure 4.13: Snapshot view of the colored point cloud, along with the ground truth central sub-aperture depth map (CSAD) are reported for each scene of the dataset `R12-ELP20`. CSAD, mask and quality map representing the absolute difference (AD) error are illustrated for all variations using the relative blur (`B`) of our framework: *top* is the scaled (`S`) coarse (`C`) and refined variations (`R`), *bottom* is the unscaled (`U`) coarse (`C`) and refined variations (`R`). Please refer to the color version for better visualization.

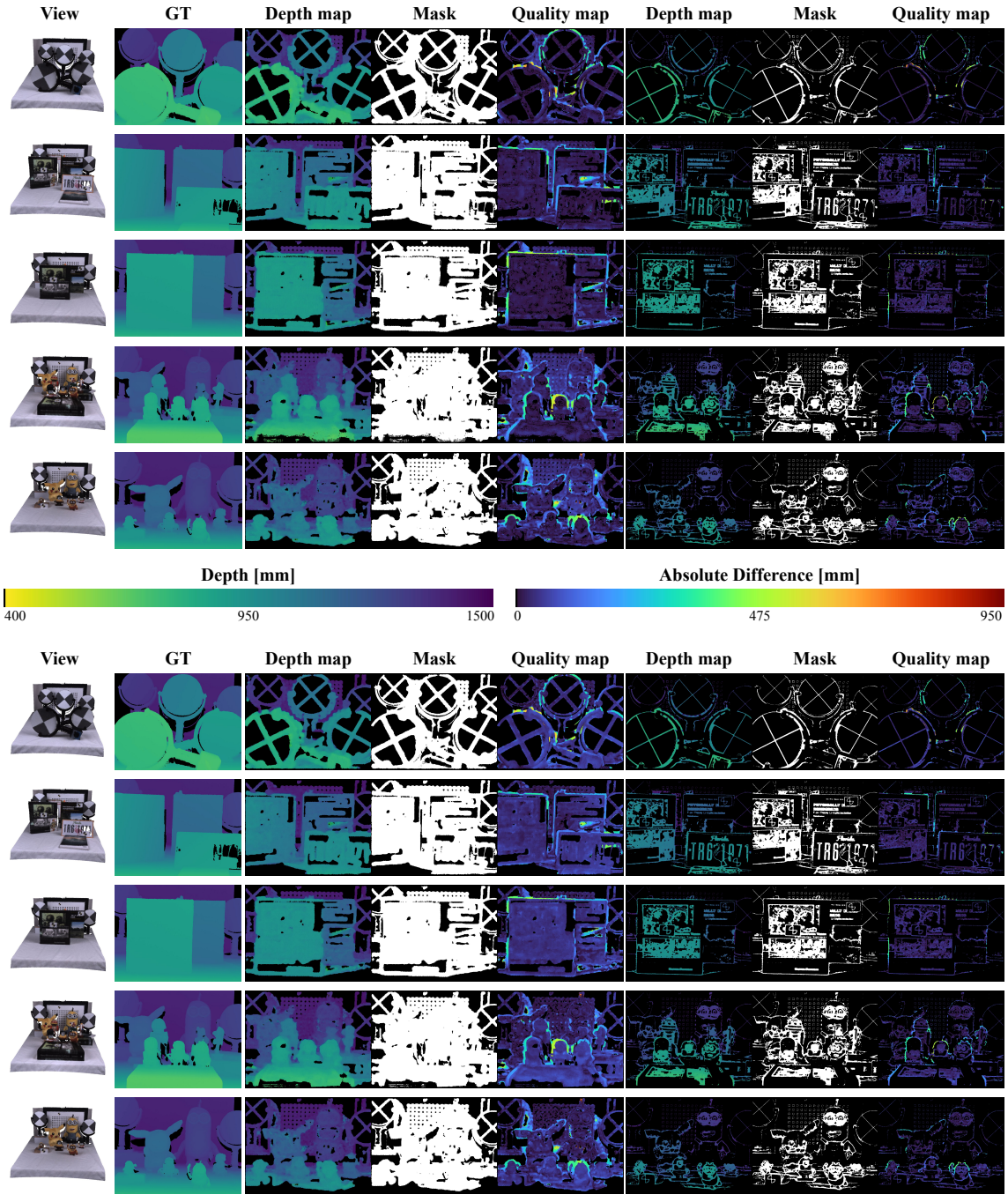


Figure 4.14: Snapshot view of the colored point cloud, along with the ground truth CSAD are reported for each scene of the dataset `R12-ELP20`. CSAD, mask and quality map representing the AD error are illustrated for all variations using only the disparity (`D`) of our framework: *top* is the scaled (`S`) coarse (`C`) and refined variations (`R`), *bottom* is the unscaled (`U`) coarse (`C`) and refined variations (`R`). Please refer to the color version for better visualization.

1. the easy scenes (`Calib`, `Plane-1` and `Plane-2`) containing mostly planar objects and presenting the lowest errors;

2. the complex scenes (`Figurines-1` and `Figurines-2`) containing more objects with less texture and more complex shapes, presenting a larger error.

First of all, the lowest overall error is obtained for the variation leveraging blur in our coarse depth estimation framework (`B/C/S`). The mean median-error over all scenes is less than 19 mm, for distances ranging from 400 mm to 1500 mm. It corresponds to relative errors ranging from 1.27 % to 4.75 % of the distance, which is coherent with the relative depth evaluation. Given the type of scene, the errors distributions are as follows:

- For easy scenes, relative errors range from 0.96 % to 3.59 % of the distance.

- For complex scenes, relative errors range from 1.73 % to 6.49 % of the distance.

As illustrated by the quality maps, most of the errors are located at objects boundaries, whereas the errors are low everywhere else. This is due to our method not explicitly dealing with occlusion boundaries, leading to wrong estimates in those regions.

Second, it is clear that the scaled variations outperform the unscaled ones. Our depth scaling calibration efficiently corrects the depth estimates, and allows to generate metric depth map without scale errors.

Third, compensating the mismatch of focus by integrating both correspondence and defocus cues shows lower median error and lower percentiles for all scenes, compared to only using disparity.

Finally, coarse estimation leads to denser depth maps as illustrated in Figure 4.13 and Figure 4.14, and is able to extrapolate depth where there is enough texture within the micro-image using a locally planar approximation. Refined estimation captures depth information only on the textured areas. The maps are sparser but wrong estimates at object boundaries do not spread as much as using the locally planar approximation.

Note that for the scene `Figurines-2`, the variation `D/R/U` has a smaller median error than the others. Recall that for `R12-E`, objects are reconstructed farther when unscaled (i.e., the scale error is negative, see Figure 4.11). As most of the errors appear on the objects' boundaries, objects in foreground are closer to the reference ones in background, and the difference of depth is then smaller. Furthermore, in the refined case, most of the estimations are done only on locally textured areas such as those boundaries. Combination of those two factors allows to explain why this variation has a smaller error.

4.4.5.5 Discussions on improvements

Sardemann and Maas [113] analyzed depth accuracy and its variance for large distance (30 m to 100 m). They concluded that focused plenoptic camera is suited for applications in mobile robotic, if we proceed to a robust filtering to eliminate outliers. Accuracy in the order of 3 % of the distance can be obtained for distance up to 100 m. Our results match their observation for our range of distances.

Computational cost is not addressed here. Our method does not operate in real-time as it is based on a purely central processing unit (CPU) implementation with a brute force algorithm for finding the minima of the cost function. We could leverage neighborhood information and implement a belief propagation strategy to avoid having to compute an initial hypothesis for each micro-image. Combined with a graphics processing unit (GPU) implementation, computation time could be significantly improved.

Furthermore, as most of the errors are located at object boundaries, we could adapt several strategies: to explicitly manage occlusions; to check coherence between estimates ref-target and target-ref; to model uncertainty as the weight function in the cost function; and we could proceed to a robust filtering to eliminate outliers, which is not done yet. Global refinement and in-painting could also be considered as further steps to improve depth estimates.

Conclusion

In this chapter, we addressed the *depth estimation* problem with plenoptic cameras, to answer the question "*How can we link image space information to the world space information?*". With a plenoptic camera, depth estimation and 3D reconstruction can be performed directly from a single acquisition, with scale information. Inherently from its design, such camera captures both *correspondences* and *defocus* cues. Both cues are complementary and can be used to estimate robust metric depth information.

We presented a new metric depth estimation algorithm using only raw images from plenoptic cameras. It is especially suited for the multi-focus configuration where several micro-lenses with different focal lengths are used. First, the main goal of our blur aware depth estimation (BLADE) approach is to improve disparity estimation for defocus stereo images via compensating the mismatch of focus, i.e., integrating both correspondence and defocus cues. We proposed to leverage blur information where it was previously considered as a drawback, by linking the relative blur to the virtual depth, i.e., the disparity. Second, we formulated a new residual error to leverage blur information for depth estimation which is used in two variations

of our framework to recover depth either at micro-image or at pixel level. Third, we showed that depth recovered from virtual depth hypothesis suffers from a scale error. We included then a depth scaling correction model as well as a methodology to calibrate it. Finally, our results show that introducing defocus cues improves the depth estimation. We demonstrated the effectiveness of our depth scaling calibration on relative depth estimation setup and on real-world 3D complex scenes with ground truths acquired with a 3D lidar scanner. With our method, we can expect a median relative error ranging from 1.27 % to 4.75 % of the distance. In our experiments it corresponds to an error of less than 19 mm, for distances ranging from 400 mm to 1500 mm.

Future work will include the discussed improvements of our method. Having a new complete camera model and enabling robust metric depth estimation from raw images only opens the door for many new applications. Further applications leveraging our contributions will be discussed in the following chapter.

Table 4.3: Intrinsic parameters for datasets `R12-A,B,C` and `R12-E` obtained by our calibration method (`BAP`), by the method of Noury *et al.* [166] (`NOUR`), and by `RxLive` software [125] (`RTRX`). They correspond to the ones used in the relative depth error evaluation of our BLADE framework.

		`R12-A` ($h = 450$ mm)			`R12-B` ($h = 1000$ mm)			`R12-C` ($h = \infty$)			`R12-E` ($h = 2133$ mm)
		`BAP`	`NOUR`	`RTRX`	`BAP`	`NOUR`	`RTRX`	`BAP`	`NOUR`	`RTRX`	`BAP`
F	[mm]	49.885	54.761	47.709	50.011	51.177	50.894	50.099	51.644	51.564	50.119
Q_1	[$\times 10^{-5}$]	24.63	6.194	-	4.661	1.650	-	13.84	1.292	-	-6.823
$-Q_2$	[$\times 10^{-6}$]	3.032	0.800	-	0.516	0.264	-	2.723	0.576	-	-0.408
Q_3	[$\times 10^{-8}$]	1.095	0.252	-	0.156	0.078	-	1.260	0.185	-	-0.047
P_1	[$\times 10^{-5}$]	-11.1	-18.1	-	12.84	11.27	-	2.51	12.13	-	20.749
$-P_2$	[$\times 10^{-5}$]	3.599	5.186	-	24.33	23.16	-	-3.072	-0.027	-	11.128
$-Q_{\text{-}1}$	[$\times 10^{-5}$]	24.29	6.195	-	4.685	1.697	-	13.78	1.316	-	-6.853
$Q_{\text{-}2}$	[$\times 10^{-6}$]	2.971	0.814	-	0.528	0.279	-	2.705	0.589	-	-0.394
$-Q_{\text{-}3}$	[$\times 10^{-8}$]	1.066	0.258	-	0.160	0.082	-	1.246	0.186	-	-0.037
$-P_{\text{-}1}$	[$\times 10^{-5}$]	-10.89	-18.00	-	12.86	11.32	-	2.50	12.17	-	21.031
$P_{\text{-}2}$	[$\times 10^{-5}$]	3.540	5.218	-	24.47	23.38	-	-3.067	-0.053	-	11.325
D	[mm]	56.860	62.341	-	52.140	53.213	-	49.356	50.728	-	50.585
$-t_x$	[mm]	10.93	9.480	-	12.15	12.38	-	12.53	13.24	-	12.876
$-t_y$	[mm]	7.996	8.087	-	6.165	5.965	-	8.237	7.400	-	6.616
$-\theta_x$	[µrad]	388.9	460.3	-	488.4	555.4	-	409.8	442.2	-	441.6
θ_y	[µrad]	271.4	363.4	-	286.5	330.1	-	306.1	333.4	-	289.2
θ_z	[µrad]	29.5	25.6	41.9	30.9	33.9	41.9	33.9	39.9	36.6	37.6
Δ_μ	[µm]	127.46	127.40	127.36	127.47	127.41	127.36	127.46	127.41	127.36	127.45
$f^{(1)}$	[µm]	582.67	-	-	566.39	-	-	580.80	-	-	601.58
$f^{(2)}$	[µm]	524.02	-	-	507.09	-	-	515.57	-	-	562.19
$f^{(3)}$	[µm]	560.57	-	-	542.47	-	-	552.84	-	-	583.54
u_0	[pix]	2078.3	2343.4	-	1855.8	1811.9	-	1786.6	1654.9	-	1722.5
v_0	[pix]	1591.0	1573.7	-	1926.2	1962.2	-	1547.1	1699.7	-	1843.6
d	[µm]	337.13	391.90	-	326.72	361.01	-	330.32	357.82	-	340.87

Chapter 5

Discussions and Perspectives

5.1 Conclusions and discussions

This thesis aimed at investigating the use of a *passive* vision sensor called a *plenoptic camera* for computer vision in robotics applications. More precisely, to achieve this goal we placed ourselves upstream of applications, and focused on its modelization to enable robust depth estimation. To answer the question "*How can we link world space information to the image space information?*", we addressed the *calibration* problem of plenoptic cameras. As a dual and more complex problem, the question "*How can we link image space information to world space information?*" has been addressed as the *depth estimation* problem with plenoptic cameras.

Calibration of plenoptic cameras. To calibrate a plenoptic camera, state-of-the-art methods rely on simplifying hypotheses, on reconstructed data or require separate calibration processes to take into account the multi-focus configuration. Taking advantage of blur information we proposed a more complete plenoptic camera model with the introduction of a new BAP feature that explicitly models the defocus blur. Our camera model tries to better fit the physical reality by describing metric quantities. It characterizes the MLA and the main lens projections within a single model, including the different micro-lenses focal lengths. The different amounts of blur within the micro-images provide a way to distinguish between them. The main

challenging part was to propose a methodology to measure the defocus blur directly in image space, and relate it to its geometric definition. This is achieved with the help of our pre-calibration step, using white raw plenoptic images. These new features are thus exploited in our calibration process based on non-linear optimization of reprojection errors. They are further leveraged in a new relative blur calibration to fill the gap between the physical and geometric blur, which enables us to fully exploit blur in image space. Our camera model is applicable to the multi-focus plenoptic camera (both in Galilean and Keplerian configuration), as well as to the single-focus and unfocused plenoptic cameras. In addition, our calibration methods have been validated by quantitative evaluations in controlled environments with real-world data. The pre-calibration step provides strong initial intrinsic parameters for the optimization. Our method showed consistent optimized camera parameters for all evaluated configurations. It presented a low and stable relative translation error across all the datasets. Thus, our model generalizes to various configurations.

Depth estimation with plenoptic cameras. To estimate depth with a plenoptic camera, state-of-the-art methods work with SAIs or EPIs. This is prone to error as depth is usually required to reconstruct the light-field or the SAIs, especially in the focused plenoptic camera case. To overcome this issue, algorithms can operate directly in the raw plenoptic images, at micro-images level. However, usually only micro-images with the smallest amount of blur are used, or alternatively, specific patterns are designed to exploit the information. In opposition, we saw that using our camera model, we relate the camera parameters to the amount of blur in the image, and all information can be used simultaneously, without distinction between types of micro-lenses. Indeed, inherently from its design, a plenoptic camera captures both *correspondences* and *defocus* cues. Both cues are complementary and can be used to estimate robust metric depth information. We presented then a new metric depth estimation algorithm using only raw images from plenoptic cameras. It is especially suited for the multi-focus configuration where several micro-lenses with different focal lengths are used. First, the main goal of our blur aware depth estimation (BLADE) approach is to improve disparity estimation for defocus stereo images via compensating the mismatch of focus, i.e., integrating both correspondence and defocus cues. We proposed to leverage blur information where it was previously considered as a drawback, by linking the relative blur to the virtual depth, i.e., the disparity. Geometric blur can be matched to the physical amount of blur in image space thanks to our relative blur calibration. Second, we formulated a new residual error to leverage blur information for depth estimation which is used in two variations of our framework to recover depth either at micro-image or at pixel level. Third, we showed that depth recovered from virtual depth hypothesis suffers from a scale error. We included then a depth scaling correction model as well as a methodology to calibrate it. Finally, our results showed that introducing defocus cues improves the depth estimation. We demonstrated the effectiveness of

our depth scaling calibration on relative depth estimation setup and on real-world 3D complex scenes with ground truth acquired with a 3D lidar scanner. With our method, we obtained accurate and precise depth estimates, with a median relative error ranging from 1.27 % to 4.75 % of the distance. In our experiments it corresponds to an error of less than 19 mm, for distances ranging from 400 mm to 1500 mm.

We believe that having a new complete camera model and enabling robust metric depth estimation from raw images only, opens the door for many new applications. It is a first step towards practical use of plenoptic cameras in computer vision applications.

5.2 Perspectives on improvements

Improvement on BAP features. Extension of our BAP features to BAP-line features could be considered. In a straight-forward fashion, the PSF could be replaced by the line-spread function (LSF), and the blur radius by the width of the blur step. Our corner detector is already able to detect micro-images containing lines (i.e., corresponding to the `border` type), and similar template-matching could be developed to find the line positional parameters. The projection model could therefore be easily adapted to project lines similar to the model of [149] but with defocus blur introduced. Reprojection of both features could be simultaneously considered, and the optimization would be more constrained as the number of observations would grow significantly.

Improvement on blur model. One current limitation of our blur model is that we operate on gray-scaled images. We loose then the chromatic information related to the `RGB` channels. However, each wavelength should provide a specific blur response. The blur proportionality coefficient could thus be calibrated for each chromatic channel. In addition, we used a simplified model for the PSF based on a Gaussian model. The choice of this model is justified as we deal with small amount of blur in the micro-images. However, future work could include the evaluation of other PSF models. This can be achieved in our framework as long as we are able to relate the geometric blur parameter to the evaluated physical model.

Improvement on depth estimation. Similarly, the similarity error measurement is purely based on gray-scaled images comparison. Use of color image should increase the discrimination between images and improve the depth estimation accuracy. It is

also common to add a term based on gradient difference to the cost function. It might be interesting to investigate the benefit of including such term in the optimization. Finally, future work should include the improvements discussed in the previous chapter, among other things, implementation on GPU, filtering, regularization, refinement and global optimization of the depth map.

5.3 Perspectives on future applications

Application to metrology of rain droplets. From an application point of view, now that we are able to relate *image* information to *world* information in a precise and accurate fashion, we want to exploit our depth estimation framework in harsh weather conditions, e.g., rain, fog or snow. In ongoing work, we are aiming at exploiting the plenoptic camera to 1) characterize the rain profile, i.e., droplet size, quantity, velocity, etc.; and 2) improve robustness of depth estimation by taking into account the droplet occlusions. It can be achieve from 3D measurements and the ability to manage small occlusions, i.e., exploiting angular information. Preliminary investigation on rain's droplet measurement shows that depth can be estimated for the droplets, and 3D reconstruction achieved, as illustrated in Figure 5.1. However, depth can be retrieved mainly due to the texture generated by reflection effects. It is not clear yet if refraction effects are exploitable or not, but there is room for improvements. Acquisition with ground truth can be realized thanks to the rain simulator of the Cerema (Centre d'Études et d'expertise sur les Risques, l'Environnement, la Mobilité et l'Aménagement), in a similar setup as the one shown in Figure 5.2. Similarly, application to PIV can also be considered.

Application to robotics. Other future works could include the exploitation of our BAP features for visual servoing, or of the point cloud representation for robotics applications, such as 3D reconstruction, 3D mapping, localization (for instance, based on the Iterative Closest Point (ICP) algorithm [236], [237]), SLAM [110], [238], etc.

Extension to monocentric lens-based plenoptic camera. One limitation of the plenoptic camera is its limited FoV (approximately 20°), whereas conventional cameras can reach angles larger than 180° (e.g., using fisheyes lenses or mirror-based catadioptric systems). Such lenses or systems cannot be considered for plenoptic camera due to their small aperture and their large f-number [239]. A new wide-field-of-view plenoptic camera system (up to 120°) has been proposed based on

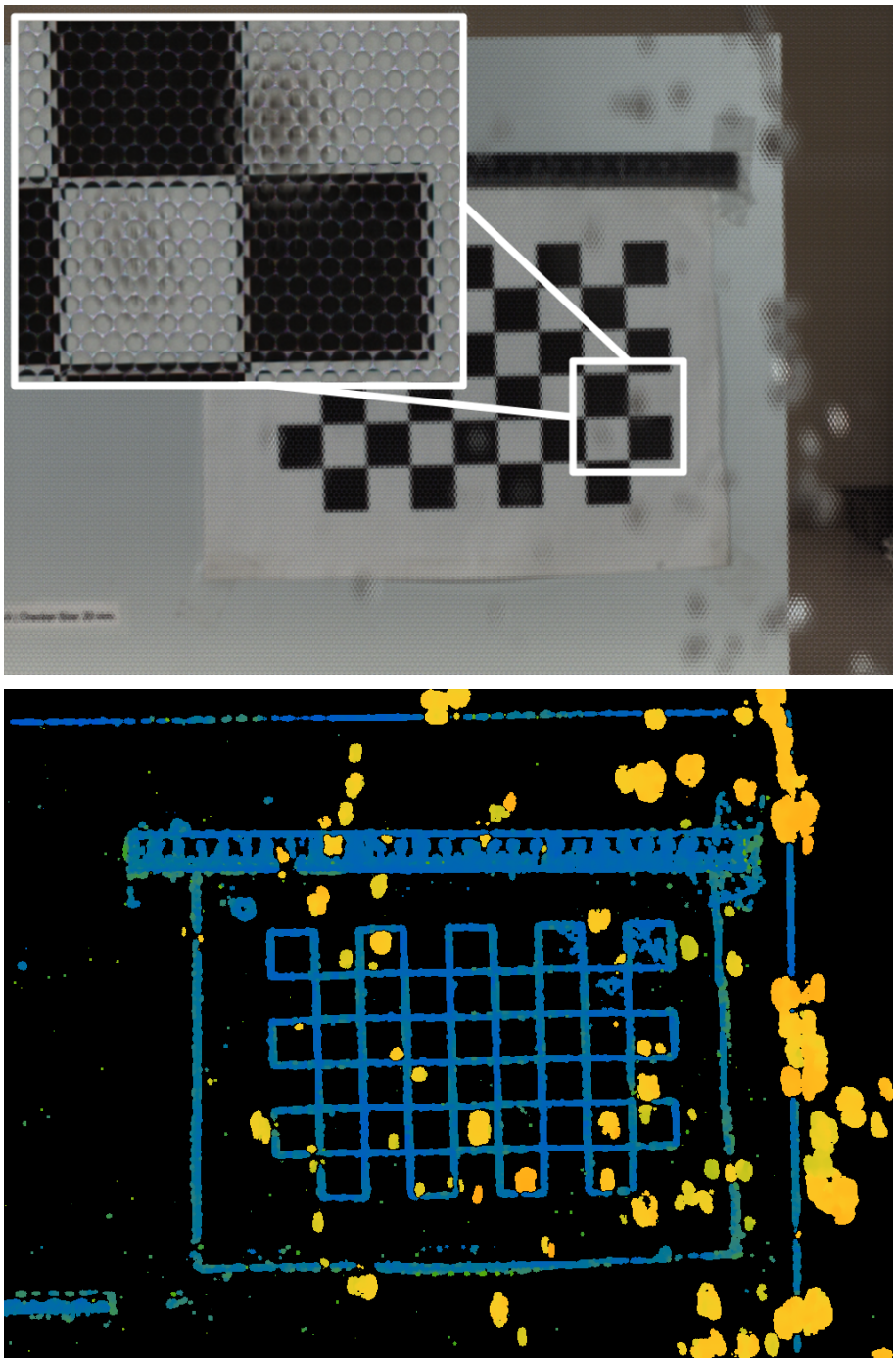


Figure 5.1: Example of raw plenoptic image with the presence of rain droplets, along with the reconstructed depth map. Depth can be retrieved mainly due to the texture generated by reflection effects. In our preliminary investigation, the depth map is obtained with the `RxLive` software.

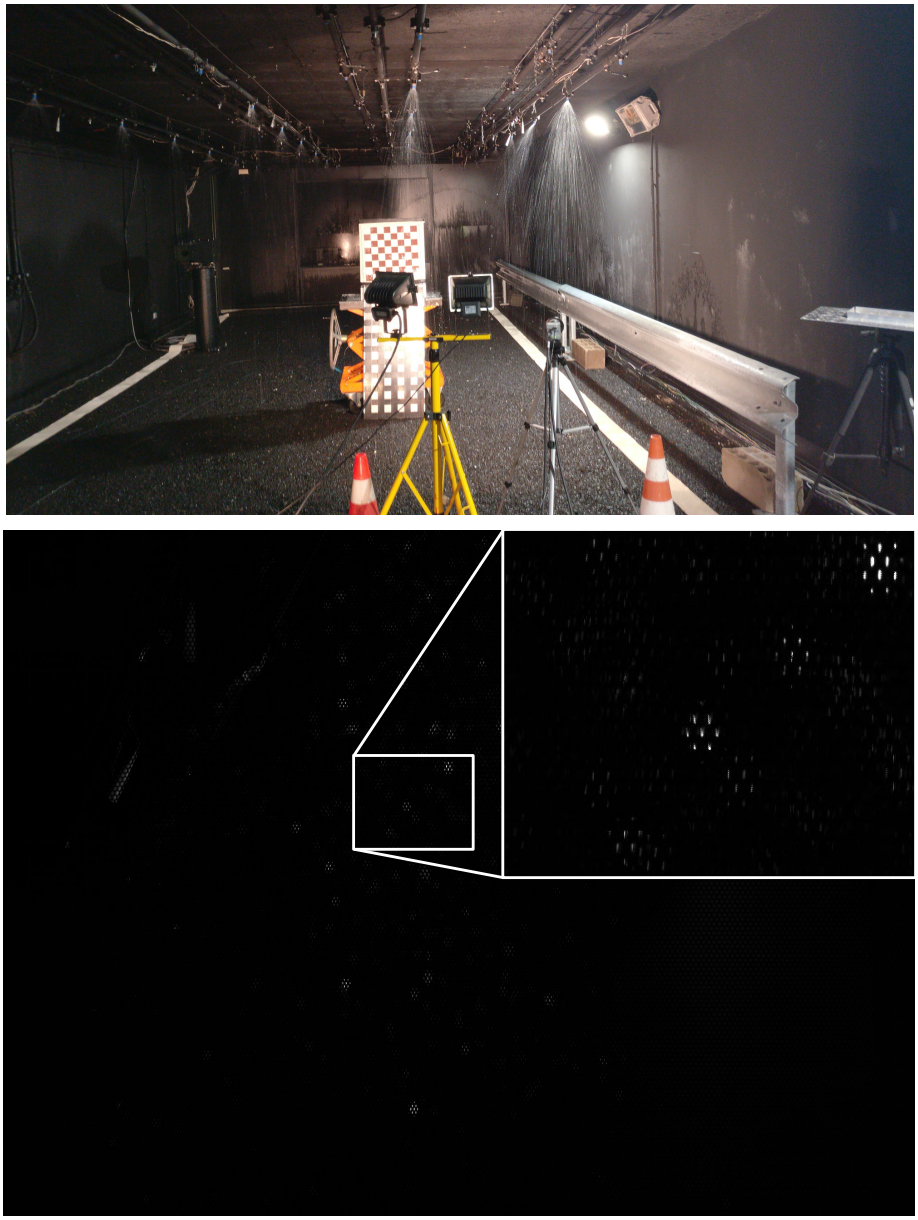


Figure 5.2: Our experimental setup with our plenoptic camera in the rain simulator of the Cerema to measure rain droplets, along with a raw plenoptic image of the scene.

monocentric lens [240]. Extension of our camera model to such system could be considered.

Application to co-design of plenoptic camera and depth estimation. In another direction, computational imaging can be considered to develop a *co-design* approach [241] applied to our camera model. Co-design aims to design simultaneously the optics parameters and the desired application, e.g., image quality restoration or depth estimation, such as for instance in an end-to-end optimization fashion [242].

Appendices

Appendix A

Publications and communications

The contributions of this thesis have been published in an international conference, submitted to two international journals, and presented in national workshops and conferences.

International Proceedings and Journals

Work related to the calibration of plenoptic cameras has been published as

> [13] M. Labussière *et al.*, "Blur Aware Calibration of Multi-Focus Plenoptic Camera", in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2020, pp. 2542–2551 **(accepted for oral presentation)**

and has been extended as a journal version as

> [14] M. Labussière *et al.*, "Leveraging blur information for plenoptic camera calibration", ***under revision*** *in International Journal of Computer Vision (IJCV)*, pp. 1–22, 2021

Work related to depth estimation with plenoptic cameras has been submitted as a journal version as

> [15] M. Labussière *et al.*, "Blur Aware Depth Estimation with a Plenoptic Camera", ***submitted to*** *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, pp. 1–17, 2022

Proceedings & Workshops without act

Other communications of this work include a poster presentation on overview of the plenoptic camera and its applications in robotics as

> M. Labussière *et al.*, "Plenoptic Cameras for Localization in Challenging Weather Conditions", *Journée Scientifique de l'École Doctorale Sciences Pour l'Ingénieur* (JS-EDSPI), May, 2019.

Calibration work has been accepted for communication not part of the act in the national conference "*Reconnaissance des Formes, Image, Apprentissage et Perception*" (RFIAP), June 2020, conjointly held with "*Conférence sur l'Apprentissage automatique*" (CAp).

The whole of this work has also been presented in a national workshop without act as

> M. Labussière *et al.*, "Leveraging Blur Information with a Plenoptic Camera: Calibration, Relative Blur calibration and characterization", *Journée thématique GdR ISIS - Capteurs visuels émergents : vision plénoptique*, Nov., 2020.

Appendix B

Datasets

B.1 R12-A, B, C

We have presented three datasets in our paper [13]: `R12-A`, `R12-B`, and `R12-C`. The devignetted images of the calibration targets from the dataset `R12-A` (Figure B.1), `R12-B` (Figure B.3), and `R12-C` (Figure B.5) taken at various angles and distances are presented below along with the poses at which they have been taken (Figure B.2, Figure B.4, and Figure B.6). The datasets can be downloaded from https://drive.uca.fr/f/d3a73cb1926047a8b635/?dl=1.

B.1.1 Experimental setup

For all experiments we used a `Raytrix R12` color 3D-light-field-camera, with a MLA of F/2.4 aperture. The camera is in Galilean *internal* configuration. The mounted lens is a `Nikon AF Nikkor F/1.8D` with a 50 mm focal length. The MLA organization is hexagonal row-aligned, and composed of 176×152 (width $\times$ height) micro-lenses with $I = 3$ different types. The sensor is a `Basler beA4000-62KC`[1] with a pixel size of $s = 0.0055$ mm. The raw image resolution is 4080×3068. All images has been acquired using the free software `MultiCam Studio` (`v6.15.1.3573`) of the company `Euresys`[2]. The shutter speed has been set to 5 ms. While taking white images for the pre-calibration step, the gain has been set to its maximum value. For `Raytrix`[3] data, we use their proprietary software `RxLive` (`v4.0.50.2`) to calibrate the camera, and compute the depth maps used in the evaluation.

B.1.2 Datasets

Each dataset is composed of:

- white raw plenoptic images acquired at different apertures ($N \in \{4, 5.66, 8, 11.31, 16\}$) using a light diffuser mounted on the main objective for pre-calibration,
- free-hand calibration target images acquired at various poses (in distance and orientation), separated into two subsets, one for the calibration process (16 images) and the other for reprojection error evaluation (15 images),
- a white raw plenoptic image acquired in the same luminosity condition and with the same aperture as in the calibration targets acquisition for devignetting,
- and, calibration targets acquired with a controlled translation motion for quantitative evaluation, along with the depth maps computed by the `RxLive` software.

B.1.2.1 Dataset R12-A

The dataset has been taken at short focus distance, $h = 450$ mm. We used a 9×5 checkerboard. Therefore, the checkerboard squares size had to be decreased to 10 mm so we can observe the corner in image space. All the poses have been acquired at distances between 400 and 175 mm from the checkerboard.

[1] https://www.baslerweb.com/en/products/cameras/area-scan-cameras/basler-beat/bea4000-62kc/ (last visited 09 Nov. 2021)

[2] https://www.euresys.com/en/Homepage

[3] https://raytrix.de/

Controlled evaluation. The dataset is composed of 11 poses taken with a relative step of 10 mm between each pose along the z-axis direction, at distances between 385 and 265 mm.

B.1.2.2 Dataset R12-B

The dataset has been taken at middle focus distance, $h = 1000$ mm. We used a 8×5 checkerboard. Therefore, the checkerboard squares size is set to 20 mm so that we can observe the corners in image space. All the poses have been acquired at distances between 775 and 400 mm from the checkerboard.

Controlled evaluation. The dataset is composed of 10 poses taken with a relative step of 50 mm between each pose along the z-axis direction, at distances between 900 and 450 mm.

B.1.2.3 Dataset R12-C

The dataset has been taken at long focus distance, $h = \infty$. We used a 6×4 checkerboard. Therefore, the checkerboard squares size had to be increased to 30mm so that we can observe the corner in image space. All the poses have been acquired at distances between 2500 and 500 mm from the checkerboard.

Controlled evaluation. The dataset is composed of 18 poses taken with a relative step of 50 mm between each pose along the z-axis direction, at distances between 1250 and 400 mm.

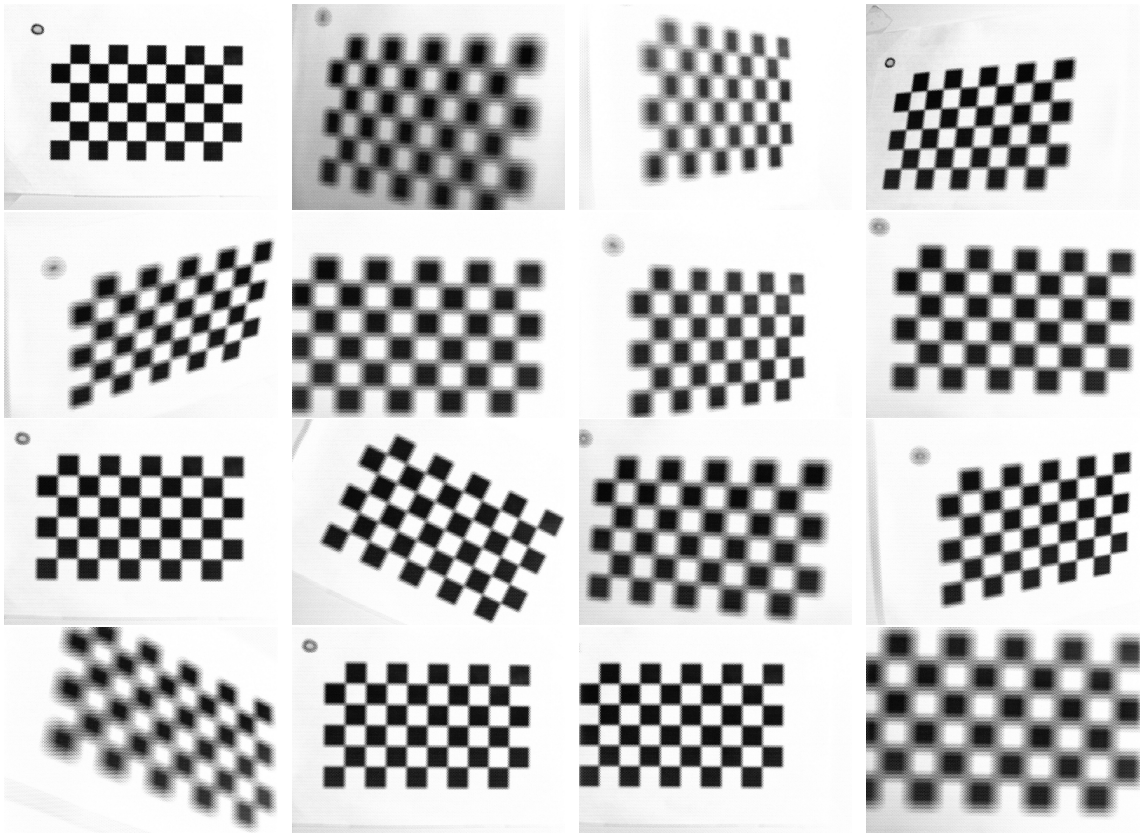


Figure B.1: Devignetted images of the calibration targets (9×5 of 10 mm side checkerboard) from the dataset R12-A taken at various angles and distances.

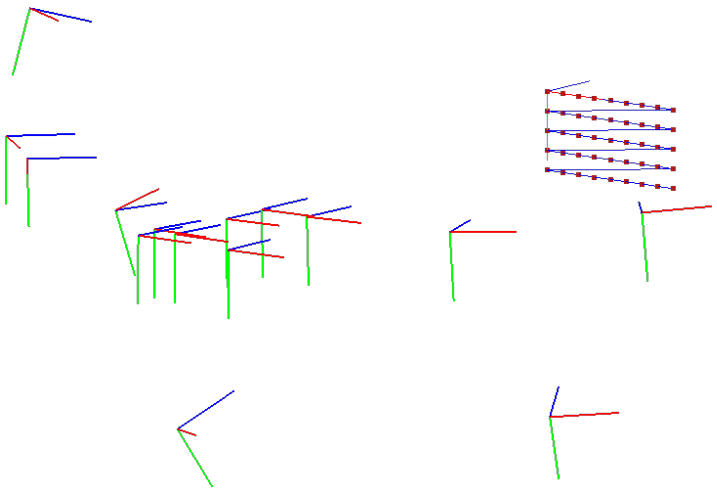


Figure B.2: Poses of the camera while capturing the calibration targets from dataset R12-A.

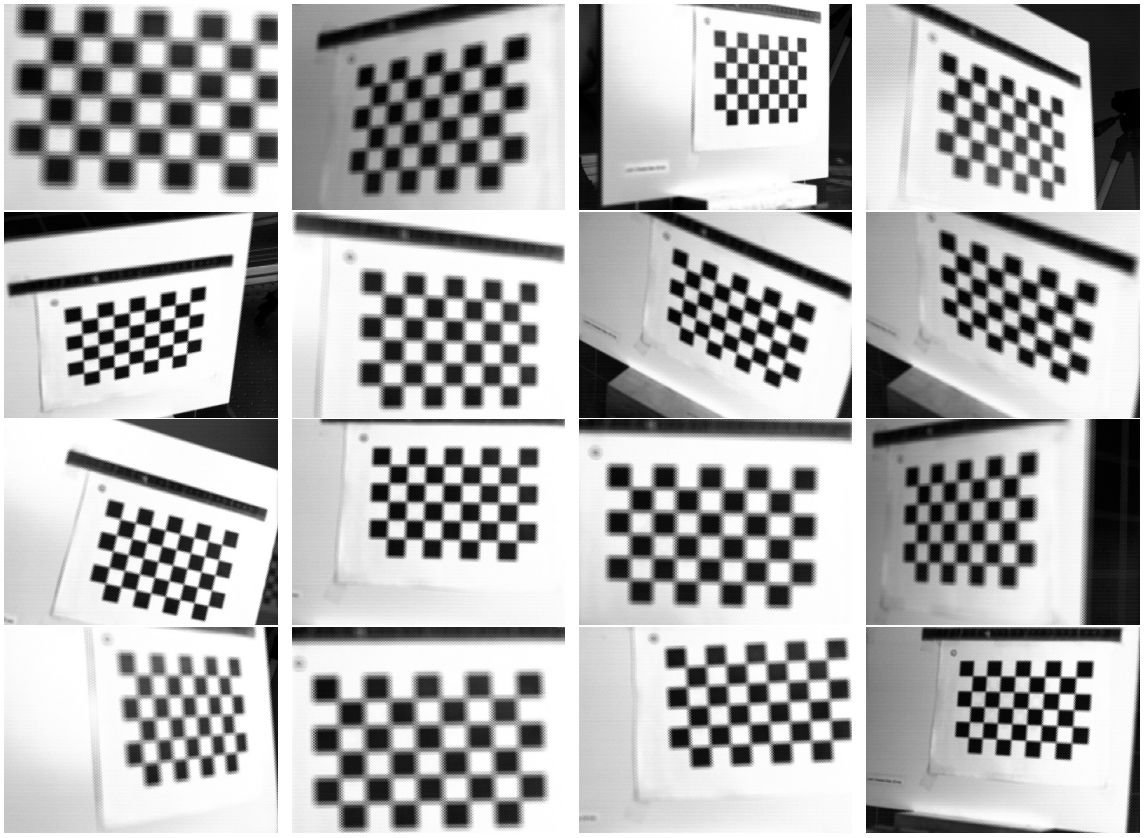


Figure B.3: Devignetted images of the calibration targets (8×5 of 20 mm side checkerboard) from the dataset R12-B taken at various angles and distances.

Figure B.4: Poses of the camera while capturing the calibration targets from dataset R12-B.

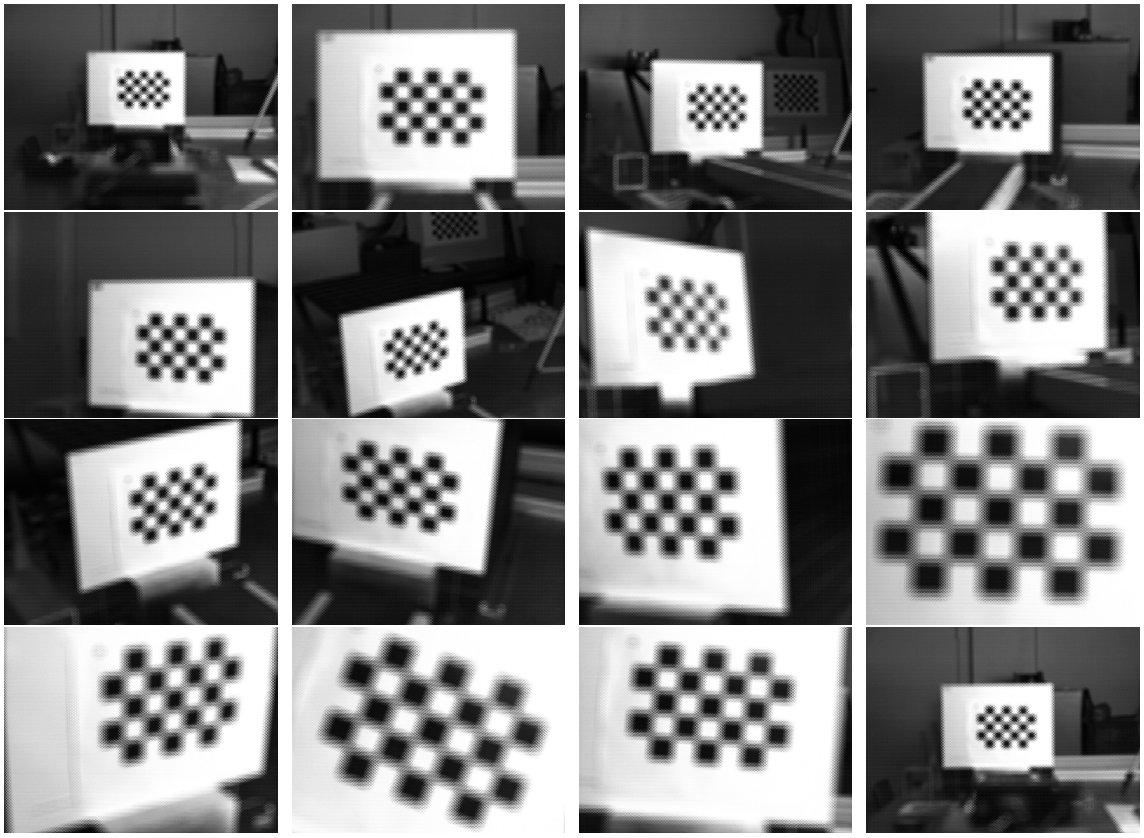


Figure B.5: Devignetted images of the calibration targets (6×4 of 30 mm side checkerboard) from the dataset R12-C taken at various angles and distances.

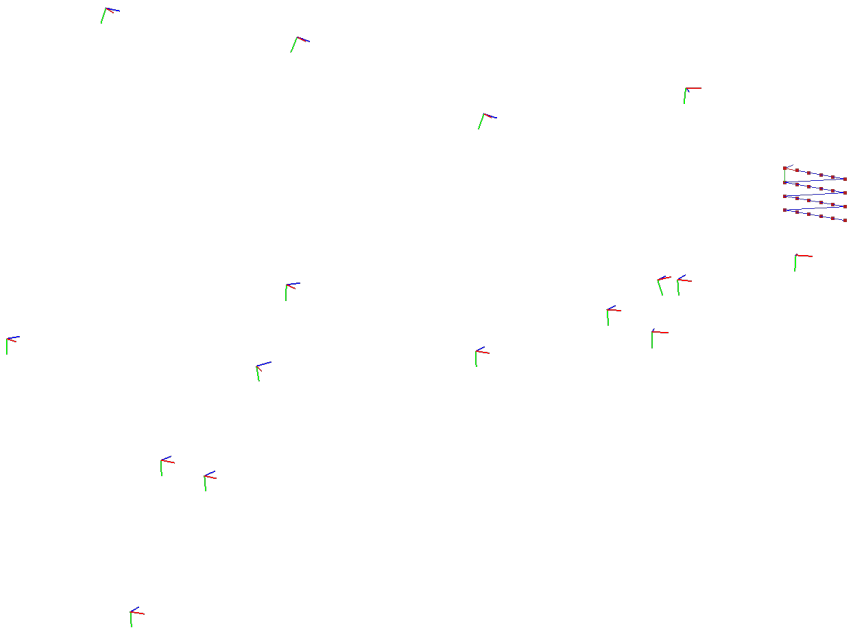


Figure B.6: Poses of the camera while capturing the calibration targets from dataset R12-C.

B.2 R12-D

We extended the previous three datasets, by introducing a new dataset in our paper [14], corresponding to another main lens configuration. The dataset can be downloaded from https://drive.uca.fr/f/bde8b32c892243ff95c4/?dl=1.

B.2.1 Experimental setup

For all experiments we used a `Raytrix R12` color 3D-light-field-camera, with a MLA of F/2.4 aperture. The camera is in Galilean *internal* configuration. The mounted lens is a `Nikon AF DC-Nikkor F/2D` with a 135 mm focal length. The MLA organization is hexagonal row-aligned, and composed of 176×152 (width $\times$ height) micro-lenses with $I = 3$ different types. The sensor is a `Basler beA4000-62KC` with a pixel size of $s = 0.0055$ mm. The raw image resolution is 4080×3068. All images has been acquired using the free software `MultiCam Studio` (`v6.15.1.3573`) of the company `Euresys`. The shutter speed has been set to 5 ms. While taking white images for the pre-calibration step, the gain has been set to its maximum value. For `Raytrix` data, we use their proprietary software `RxLive` (`v4.0.50.2`) to calibrate the camera, and compute the depth maps used in the evaluation.

Figure B.7: Our `Raytrix R12` plenoptic camera with the mounted lens of 135 mm focal length used in dataset `R12-D`.

B.2.2 Datasets

The dataset corresponds to the focus distance configuration $h = 1500$ mm. We use a 5×3 of 20 mm side checkerboard. Each dataset is composed of:

- white raw plenoptic images acquired at different apertures ($N \in \{2.8, 4, 5.66, 8, 11.31, 16\}$) using a light diffuser mounted on the main objective for pre-calibration,
- free-hand calibration target images acquired at various poses (in distance and orientation), separated into two subsets, one for the calibration process (16 images) and the other for reprojection error evaluation (15 images),
- a white raw plenoptic image acquired in the same luminosity condition and with the same aperture as in the calibration targets acquisition for devignetting,
- and, calibration targets acquired with a controlled translation motion for quantitative evaluation, along with the depth maps computed by the `RxLive` software.

Controlled evaluation. The dataset is composed of 13 poses taken with a relative step of 50 mm between each pose along the z-axis direction, at distances between 1200 and 750 mm.

B.3 UPC-S

We also presented a simulated dataset for unfocused plenoptic camera, i.e., `Lytro`-like plenoptic camera configuration, in our paper [14]. The dataset can be downloaded from https://drive.uca.fr/f/c617039b1dd14ad78e84/?dl=1.

B.3.1 Experimental setup

We used the `Lytro Illum` intrinsic parameters reported in [149, Table 4] as baseline for the simulation, corresponding to a main lens of aperture $F/2$ with a 9.9845 mm focal length. The camera is in unfocused *internal* configuration (i.e., $f = d$). The MLA organization is hexagonal row-aligned, and composed of 541×434 (width $\times$ height) micro-lenses of the same type (i.e., $I = 1$). The raw image resolution is 7728×5368 pixel, with a pixel size of $s = 0.0014$ mm and with micro-image of radius 7.172 pixel. All images has been generated using the `libpleno` and our raytracing simulator `prism` using 1500 rays per pixel.

B.3.2 Dataset

The dataset is correspond to the focus distance configuration h = hyperfocal. We use a 9×5 of 26.25 mm side checkerboard, as in the reference setup of [149]. The dataset is composed of:

- white raw plenoptic images simulated at different apertures ($N \in \{2, 4, 5.6\}$) for pre-calibration step,
- free-hand calibration targets (23 images) simulated at various poses (in distance and orientation, between 250 and 800 mm) for the calibration process,
- and calibration targets with known translation along the z-axis for quantitative evaluation.

Controlled evaluation. The dataset is composed of 7 poses taken with a relative step of 50 mm between each pose along the z-axis direction, at distances between 200 and 500 mm.

B.4 R12-E, ES, ELP20

Finally, we presented a last dataset containing ground truth data on 3D complex real-world scene in our paper [15]. The dataset can be downloaded from `https://drive.uca.fr/f/f164345e148642b881c3/?dl=1`.

B.4.1 Experimental setup

For our experiments we used a `Raytrix R12` color 3D-light-field-camera, with a MLA of F/2.4 aperture. The camera is in Galilean *internal* configuration. The mounted lens is a `Nikon AF Nikkor F/1.8D` with a 50 mm focal length. The MLA organization is hexagonal row-aligned, and composed of 176×152 (width $\times$ height) micro-lenses with $I = 3$ different types. The sensor is a `Basler beA4000-62KC` with a pixel size of $s = 0.0055$ mm. The raw image resolution is 4080×3068 pixel. All images have been acquired using the `MultiCamStudio` free software (v6.15.1.3573) of the `Euresys` company. We set the shutter speed to 5 ms.

3D scenes setup. We use a 3D lidar scanner, a `Leica ScanStation P20 (LP20)`, that allows us to capture a color point cloud with high precision that can be used as

ground truth data. The `LP20` is configured with no HDR and with a resolution of 1.6 mm @ 10 m.

B.4.2 Datasets

The configuration corresponds to a focus distance $h = 2133$ mm. We built a calibration dataset, using a 6×4 of 30 mm side checkerboard, which is composed of:

- white raw plenoptic images acquired at different apertures ($N \in \{2.8, 4, 5.66, 8, 11.31, 16\}$) using a light diffuser mounted on the main objective for pre-calibration,
- free-hand calibration target images acquired at various poses (in distance and orientation) for the calibration process (31 images),
- a white raw plenoptic image acquired in the same luminosity condition and with the same aperture as in the calibration targets acquisition for devignetting.

With this configuration, we created two sub-datasets:

1. a simulated dataset built upon our own simulator `prism` based on raytracing to generate images (with 1500 rays/pixel) with known absolute position for quantitative evaluation, named `R12-ES` (15 images, from 500 mm to 1900 mm with a step of 100 mm);
2. a dataset composed of several 3D scenes with ground truth acquired with the `LP20`, for object distances ranging from 400 mm to 1500 mm.

The latter dataset, named `R12-ELP20`, includes fives scenes:

- one scene for extrinsic parameters calibration, containing checker corner targets, named `Calib`;
- two scenes containing textured planar objects, named `Plane-1` and `Plane-2`;
- and two more complex scenes containing various figurines, named `Figurines-1` and `Figurines-2`.

Each scene is composed of: a colored point cloud (with spatial (x, y, z) information, color information (r, g, b), and intensity information) in format `.ptx`, `.pts` and `.xyz`; 3D positions of the targets in the lidar reference frame; two raw plenoptic images in `rgb` color and two raw plenoptic images in bayer; finally, photos and labels of the scene.

Appendix C

Source code

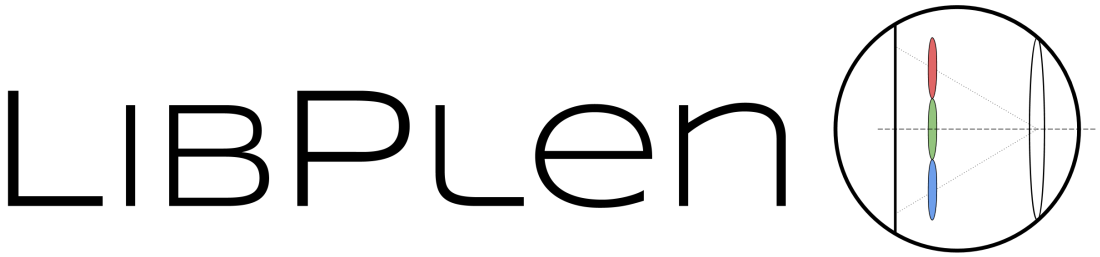
All our code sources have been made publicly available on the lab's GitHub page, https://github.com/comsee-research, for reproducibility and broad accessibility, and licensed under the GNU General Public License v3.0. It includes:

`libpleno` available at https://github.com/comsee-research/libpleno.

`compote` available at https://github.com/comsee-research/compote.

`prism` available at https://github.com/comsee-research/prism.

`blade` available at https://github.com/comsee-research/blade.



C.1 libpleno

The `libpleno` is an open-source `C++` computer-vision library for plenoptic cameras modeling and processing. It has a light dependency list:

eigen version 3, a modern C++ matrix and linear-algebra library,

boost version 1.54 and up, portable C++ source libraries,

opencv version 3.2, a collection of algorithms and sample code for various computer vision problems,

libv a general purpose computer vision library developed at Pascal Institute, used mostly for graphic and serialization,

lma a non-linear optimization library implementing the Levenberg Marquardt Algorithm,

and was compiled on: Ubuntu `18.04.4 LTS`, with `C++17`, with GCC `7.5.0`, with Eigen `3.3.4`, Boost `1.65.1`, and OpenCV `3.2.0`. If the reader is comfortable with Linux and CMake and has already installed the prerequisites above, the following commands should install the `libpleno` on your system.

```
mkdir build && cd build
cmake .. -DUSE_OPEN_MP=true
make -j6
sudo make install
```

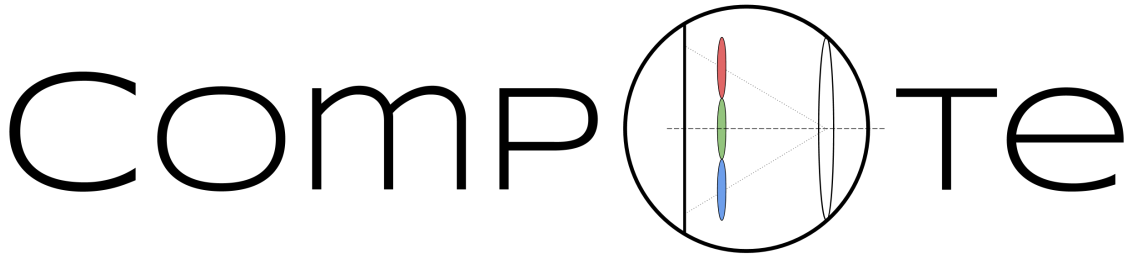
Once installed the user can use the `libpleno` by completing the `CMakeLists.txt` of the apps with

```
find_package(libpleno REQUIRED)
```

an using the defined variables

```
${LIBPLENO_LIBRARIES}
${LIBPLENO_INCLUDE_DIRS}
```

Some examples of configuration files are included in the repository.



C.2 compote

The project `compote`, for Calibration Of Multi-focus PlenOpTic camEra, is a set of tools to pre-calibrate and calibrate (multi-focus) plenoptic cameras based on the `libpleno`.

Configuration. All applications use `.js` (json) configuration file. The path to this configuration files are given in the command line using `boost program options` interface. Options are:

short	long	default	description
-h	--help		Print help messages
-g	--gui	'true'	Enable GUI (image viewers, etc.)
-v	--verbose	'true'	Enable output with extra information
-l	--level	`ALL` (15)	Select level of output to print (can be combined): `NONE`=0, `ERR`=1, `WARN`=2, `INFO`=4, `DEBUG`=8, `ALL`=15
-i	--pimages		Path to images configuration file
-c	--pcamera		Path to camera configuration file
-p	--pparams	'internals.js'	Path to camera internal parameters configuration file
-s	--pscene		Path to scene configuration file
-f	--features	'observations.bin.gz'	Path to observations file
-e	--extrinsics	'extrinsics.js'	Path to save extrinsics parameters file
-o	--output	'intrinsics.js'	Path to save intrinsics parameters file

For instance to run calibration:

```
./calibrate -i images.js -c camera.js -p params.js \
            -f observations.bin.gz -s scene.js -g true -l 7
```

Some examples of configuration files are included in the repository.

Applications. Five applications are included in `compote`:

`precalibrate` uses whites raw images taken at different apertures to calibrate the micro-images array (MIA) and computes the *internal parameters* used to initialize the camera and to detect the blur aware plenoptic (BAP) features.

Requirements: minimal camera configuration, white images.
Output: radii statistics (.csv), internal parameters, initial camera parameters.

`detect` extracts the newly introduced BAP features in checkerboard images.

Requirements: calibrated MIA, computed internal parameters, checkerboard images, and scene configuration.
Output: micro-image centers and BAP features.

`calibrate` runs the calibration of the plenoptic camera. Set $I = 0$ to act as pinholes array, or $I > 0$ for the multi-focus case.

Requirements: calibrated MIA, internal parameters, features and scene configuration. If none are given all previous steps are re-done.
Output: error statistics, calibrated camera parameters, camera poses.

`extrinsics` runs the optimization of extrinsics parameters given a calibrated camera and estimates the poses.

Requirements: internal parameters, features, calibrated camera and scene configuration.
Output: error statistics, estimated poses.

`blur` runs the calibration of the blur proportionality coefficient κ linking the spread parameter of the PSF with the blur radius. It updates the internal parameters with the optimized value of κ.

Requirements: internal parameters, features and images.
Output: internal parameters.

`invdistortion` runs the calibration of the inverse distortion coefficients ϕ^{-1} used in the inverse projection model.

Requirements: camera parameters, internal parameters and scene configuration.
Output: calibrated camera parameters.

It also provides two legacy applications to run statistics evaluation on the optimized poses obtained with a constant step linear translation along the z-axis:

`linear_evaluation` gives the absolute errors (mean + std) and the relative errors (mean + std) of translation of the optimized poses,

`linear_raytrix_evaluation` takes `.xyz` point clouds obtained by `Raytrix` calibration software and gives the absolute errors (mean + std) and the relative errors (mean + std) of translation.

Note: those apps are legacy and have been moved and generalized in the `blade` app's `evaluate`. If you want to enable the compilation of legacy applications for evaluations, add the option `-DCOMPILE_LEGACY_EVAL` to `cmake`.

C.3 prism

The project `prism`, for Plenoptic Raw Image Simulator, is a set of tools to generate and simulate raw images from (multi-focus) plenoptic cameras based on the `libpleno`.

Configuration. Options passed in command line using `boost program options` interface are

short	long	default	description
-h	--help		Print help messages
-g	--gui	‘true‘	Enable GUI (image viewers, etc.)
-v	--verbose	‘true‘	Enable output with extra information
-l	--level	`ALL` (15)	Select level of output to print (can be combined): `NONE`=0, `ERR`=1, `WARN`=2, `INFO`=4, `DEBUG`=8, `ALL`=15
-c	--pcamera		Path to camera configuration file
-s	--pscene		Path to scene configuration file
-n	--nrays	‘30‘	Number of rays per pixel
	--vignetting	‘true‘	Enable vignetting effect in modelization
	--run_all	‘false‘	Run automaticaly all image generation
	--save_all	‘false‘	Save automaticaly all image
-n	--nposes	‘10‘	Number of poses to generate
	--min	‘450‘	Distance min for pose generation
	--max	‘1900‘	Distance max for pose generation

For instance to run images generation:

```
./src/prism/prism -s scene.js -c camera.js --nrays 30 \
                  --vignetting false --run_all true --save_all true \
                  -v true -g true -l 7
```

To test the `prism` application you can use the example script from the build directory:

```
./../examples/run_prism.sh
```

Some examples of configuration files are included in the repository.

Applications. Two applications are included in `prism`:

`prism` generates images based on raytracing according to the scene configuration.

> **Requirements:** camera parameters, scene configuration and number of rays per pixel.
> **Output:** images.

`scene` generates randomly valid poses and the scene configuration.

> **Requirements:** min/max depths, number of poses, camera parameters, scene configuration and texture.
> **Output:** poses and scene configuration.

C.4 blade

The project `blade`, for BLur Aware Depth Estimation, is a set of tools to estimate depth map from raw images obtained by (multi-focus) plenoptic cameras based on the `libpleno`.

Configuration. All applications use `.js` (json) configuration file. The path to this configuration files are given in the command line using `boost program options` interface.

For instance to run depth estimation:

```
./depth -i images.js -c camera.js -p params.js \
        -o depth.png -v true -g true -l 7
```

To test the `blade` application you can use the example script from the build directory:

```
./../examples/depth.sh
```

Options are:

short	long	default	description
-h	–help		Print help messages
-g	--gui	‘true‘	Enable GUI (image viewers, etc.)
-v	--verbose	‘true‘	Enable output with extra information
-l	--level	`ALL` (15)	Select level of output to print (can be combined): `NONE`=0, `ERR`=1, `WARN`=2, `INFO`=4, `DEBUG`=8, `ALL`=15
-i	--pimages		Path to images configuration file
-c	--pcamera		Path to camera configuration file
-p	--pparams	‘internals.js‘	Path to camera internal parameters configuration file
-s	--pscene		Path to scene configuration file
-f	--features	‘observations.bin.gz‘	Path to observations file
-e	--extrinsics	‘extrinsics.js‘	Path to save extrinsics parameters file
-o	--output	‘intrinsics.js‘	Path to save intrinsics parameters file

Applications. Five applications are included in `blade`:

`depth` runs depth estimations on input images according to the selected strategy. In particular, `depth_from_obs` runs depth estimations on input images according to the selected strategy at micro-images containing BAP features only.

Requirements: image(s), camera parameters, internal parameters, strategy configuration.
Output: raw depth map(s), point cloud(s), central sub-aperture depth map(s).

`scaling` runs the depth scaling calibration process.

Requirements: images, camera parameters, internal parameters, scene configuration, raw depth maps, features.
Output: camera parameters, scale error statistics (`.csv`).

`evaluate` runs the evaluations of relative depth estimation with respect to a ground truth. Supported depth formats include: raw depth maps, point clouds, `.csv`, `.pts`, `.xyz`, poses, `.mat` and planes.

Requirements: camera parameters, internal parameters, ground truth and depth information.
Output: absolute and relative errors statistics (`.csv`).

`lidarcamera` runs the extrinsic parameters calibration from lidar frame to camera frame, and graphically check the point clouds.

Requirements: camera parameters, internal parameters, calibration image, constellation configuration.
Output: extrinsic parameters.

`distances` evaluates distances between reference point cloud and computed depth information, either, directly from central sub-aperture depth map(s) or point cloud(s) or raw depth map(s).

Requirements: camera parameters, internal parameters, extrinsic lidar-camera parameters, images, reference point cloud (`.pts`), depth information to evaluate.
Output: error maps, distances.

Appendix D

Quantification of the approximation Eq. (4.9)

Let $a = 2nd$ be the distance to the micro-lens (i), with $n \in [1, 10]$, such that $\upsilon = a/d = 2n$. Let $\delta z = nB\sin(\alpha)$ be the z-shift between the micro-lenses (i) and (j) separated by a distance nB. The virtual depth υ' for the micro-lens (j) is then given by $\upsilon' = (a + \delta z) / (d + \delta z)$.
First, as $B < d/2$, and with $\sin(\alpha) \approx \alpha$ since α is small, we can express υ' as

$$\upsilon' = \frac{a + \delta z}{d + \delta z} = \frac{a + \alpha nB}{d + \alpha nB} < \frac{2nd + \alpha\frac{n}{2}d}{d + \alpha\frac{n}{2}d} < \frac{2 + \frac{\alpha}{2}}{\frac{1}{n} + \frac{\alpha}{2}}. \tag{D.1}$$

Second, let us measures the approximation error of υ' by υ. The relative error is given by

$$\begin{aligned} \varepsilon &= \left|1 - \frac{\upsilon}{\upsilon'}\right| = \left|1 - 2n \cdot \frac{1/n + \alpha/2}{2 + \alpha/2}\right| \\ &= \left|1 - \frac{2 + n\alpha}{2 + \alpha/2}\right| = \left|\frac{\alpha/2 - n\alpha}{2 + \alpha/2}\right|. \end{aligned} \tag{D.2}$$

Finally, from calibration, $\alpha < 0.0005$ rad, the relative error is thus bounded such that $0.0125\ \% < \varepsilon_{\%} < 0.2375\ \%$, which shows the approximation is valid.

□

Bibliography

[1] F. E. Ives, "Parallax Stereogram and Process of making same", *US Patent 725,567*, pp. 1–3, 1903.

[2] G. Lippmann, "Épreuves réversibles donnant la sensation du relief", *Journal de Physique Théorique et Appliquée*, vol. 7, no. 1, pp. 821–825, 1908.

[3] E. H. Adelson and J. Y. A. Wang, "Single Lens Stereo with a Plenoptic Camera", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 99–106, 1992.

[4] R. Ng, M. Levoy, G. Duval, M. Horowitz, and P. Hanrahan, "Light Field Photography with a Hand-held Plenoptic Camera", Stanford University, Tech. Rep., 2005, pp. 1–11.

[5] C. Perwaß and L. Wietzke, "Single Lens 3D-Camera with Extended Depth-of-Field", in *Human Vision and Electronic Imaging XVII*, vol. 49, SPIE, Feb. 2012, p. 829 108.

[6] D. G. Dansereau, "Plenoptic Signal Processing for Robust Vision in Field Robotics", PhD thesis, The University of Sydney, 2014, pp. 1–192.

[7] M. Levoy, R. Ng, A. Adams, M. Footer, and M. Horowitz, "Light field microscopy", *ACM Transactions on Graphics*, vol. 25, no. 3, pp. 924–934, 2006.

[8] F. Dong, S. H. Ieng, X. Savatier, R. Etienne-Cummings, and R. Benosman, "Plenoptic Cameras in Real-Time Robotics", *The International Journal of Robotics Research*, vol. 32, no. 2, pp. 206–217, 2013.

[9] M. Lingenauber, F. A. Fröhlich, U. Krutz, C. Nissler, and K. H. Strobl, "In-Situ Close-Range Imaging with Plenoptic Cameras", *IEEE Aerospace Conference Proceedings*, vol. 2019-March, 2019.

[10] Z. Cheng, Z. Xiong, C. Chen, and D. Liu, "Light field super-resolution: A benchmark", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, vol. 2019-June, pp. 1804–1813, 2019.

[11] T. C. Wang, A. A. Efros, and R. Ramamoorthi, "Occlusion-aware depth estimation using light-field cameras", *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015 Inter, pp. 3487–3495, 2015.

[12] M. W. Tao, P. P. Srinivasan, J. Malik, S. Rusinkiewicz, and R. Ramamoorthi, "Depth from shading, defocus, and correspondence using light-field angular coherence", in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2015, pp. 1940–1948.

[13] M. Labussière, C. Teulière, F. Bernardin, and O. Ait-Aider, "Blur Aware Calibration of Multi-Focus Plenoptic Camera", in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2020, pp. 2542–2551.

[14] ——, "Leveraging blur information for plenoptic camera calibration", ***under revision*** *in International Journal of Computer Vision (IJCV)*, pp. 1–22, 2021.

[15] ——, "Blur Aware Depth Estimation with a Plenoptic Camera", ***submitted to*** *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, pp. 1–17, 2022.

[16] G. Lippmann, "Integral Photography", *Academy of the Sciences*, 1911.

[17] A. Gershun, "The Light Field", *Journal of Mathematics and Physics*, vol. 18, no. 1-4, pp. 51–151, 1939.

[18] E. H. Adelson and J. R. Bergen, "The plenoptic function and the elements of early vision", *Computational Models of Visual Processing*, pp. 3–20, 1991.

[19] H. E. Ives, "Parallax Panoramagrams Made with a Large Diameter Lens", *Journal of the Optical Society of America*, vol. 20, no. 6, p. 332, 1930.

[20] A. Lumsdaine and T. Georgiev, "The focused plenoptic camera", in *IEEE International Conference on Computational Photography (ICCP)*, Apr. 2009, pp. 1–8.

[21] T. Georgiev and A. Lumsdaine, "Depth of Field in Plenoptic Cameras", *Eurographics 2009*, no. 1, pp. 5–8, 2009.

[22] ——, "Resolution in Plenoptic Cameras", *Frontiers in Optics 2009/Laser Science XXV/Fall 2009 OSA Optics & Photonics Technical Digest*, CTuB3, 2009.

[23] T. Georgiev, K. C. Zheng, B. Curless, D. Salesin, S. K. Nayar, and C. Intwala, "Spatio-Angular Resolution Tradeoff in Integral Photography", *Rendering Techniques*, vol. 2006, no. 263-272, p. 21, 2006.

[24] A. Levin, W. T. Freeman, and F. Durand, "Understanding camera tradeoffs through a Bayesian analysis of light field projections", *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5305 LNCS, no. PART 4, pp. 88–101, 2008.

[25] T. Georgiev and A. Lumsdaine, "The multifocus plenoptic camera", in *Digital Photography VIII*, International Society for Optics and Photonics, SPIE, 2012, pp. 69–79.

[26] 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 SIGGRAPH 2005 Papers on - SIGGRAPH '05*, New York, New York, USA: ACM Press, 2005, p. 765.

[27] K. Venkataraman, D. Lelescu, J. Duparré, A. McMahon, G. Molina, P. Chatterjee, R. Mullis, and S. K. Nayar, "PiCam", *ACM Transactions on Graphics*, vol. 32, no. 6, pp. 1–13, Nov. 2013.

[28] C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, and M. Gross, "Scene reconstruction from high spatio-angular resolution light fields", *ACM Transactions on Graphics*, vol. 32, no. 4, p. 1, Jul. 2013.

[29] C.-K. Liang, T.-H. Lin, B.-Y. Wong, C. Liu, and H. H. Chen, "Programmable aperture photography", in *ACM SIGGRAPH 2008 papers on - SIGGRAPH '08*, New York, New York, USA: ACM Press, 2008, p. 1.

[30] G. Wu, B. Masia, A. Jarabo, Y. Zhang, L. Wang, Q. Dai, T. Chai, and Y. Liu, "Light Field Image Processing: An Overview", *IEEE Journal on Selected Topics in Signal Processing*, vol. 11, no. 7, pp. 926–954, 2017.

[31] J. C. Yang, M. Everett, C. Buehler, and L. McMillan, "A real-time distributed light field camera", in *EGRW '02 Proceedings of the 13th Eurographics workshop on Rendering*, 2002, pp. 77–86.

[32] C. Zhang and T. Chen, "A self-reconfigurable camera array", *ACM SIGGRAPH 2004 Sketches on - SIGGRAPH '04*, p. 151, 2004.

[33] X. Lin, J. Wu, G. Zheng, and Q. Dai, "Camera array based light field microscopy", *Biomedical Optics Express*, vol. 6, no. 9, p. 3179, 2015.

[34] M. Levoy, "Light Fields and Computational Imaging", *Computer*, Springer Series in Chemical Physics, vol. 39, no. 8, K. Yamanouchi, Ed., pp. 46–55, Aug. 2006.

[35] T. Georgiev, Z. Yu, A. Lumsdaine, and S. Goma, "Lytro camera technology: theory, algorithms, performance analysis", vol. 5, no. 1, 86671J, 2013.

[36] C. Riou, B. Colicchio, J. P. Lauffenburger, O. Haeberlé, and C. Cudel, "Calibration and disparity maps for a depth camera based on a four-lens device", *Journal of Electronic Imaging*, vol. 24, no. 6, p. 061 108, 2015.

[37] S. Zhu, A. Lai, K. Eaton, P. Jin, and L. Gao, "On the fundamental comparison between unfocused and focused light field cameras", *Applied Optics*, vol. 57, no. 1, A1, 2018.

[38] K. Cossu, G. Druart, and M. Bonnefois, "Caméras plénoptiques pour l'imagerie tridimensionnelle", Techniques de l'Ingénieur, Tech. Rep., 2016.

[39] L. McMillan and G. Bishop, "Plenoptic modeling: An Image-Based Rendering System", in *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques - SIGGRAPH '95*, vol. 41, New York, New York, USA: ACM Press, 1995, pp. 39–46.

[40] M. Levoy and P. Hanrahan, "Light field rendering", in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques - SIGGRAPH '96*, New York, New York, USA: ACM Press, 1996, pp. 31–42.

[41] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The lumigraph", *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques - SIGGRAPH '96*, pp. 43–54, 1996.

[42] H. Schirmacher, C. Vogelgsang, H.-P. Seidel, and G. Greiner, "Efficient Free Form Light Field Rendering", *Proceedings of the Vision Modeling and Visualization Conference 2001 - VMV '01*, no. November, pp. 249–256, 2001.

[43] 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 - SIGGRAPH '00*, New York, New York, USA: ACM Press, 2000, pp. 297–306.

[44] C. Buehler, M. Bosse, L. McMillan, S. Gortler, and M. Cohen, "Unstructured lumigraph rendering", in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques - SIGGRAPH '01*, New York, New York, USA: ACM Press, 2001, pp. 425–432.

[45] M. Alain and A. Smolic, "Spectral analysis of re-parameterized light fields", *arXiv:2110.06064*, no. 15, pp. 1–37, 2021.

[46] I. Ihm, S. Park, and R. K. Lee, "Rendering of spherical light fields", *Proceedings The Fifth Pacific Conference on Computer Graphics and Applications*, pp. 59–68, 1997.

[47] E. Camahort, A. Lerios, and D. Fussell, "Uniformly Sampled Light Fields", in, 1998, pp. 117–130.

[48] G. Tsang, S. Ghali, E. L. Fiume, and A. N. Venetsanopoulos, "A novel parameterization of the light field", in *Image and Multidimensional Digital Signal Processing'98 (Proc. 10th IMDSP Workshop)*, 1998.

[49] 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*, vol. 1, no. 1, pp. 7–55, 1987.

[50] S. C. Chan and H.-Y. Shum, "A spectral analysis for light field rendering", *2000 International Conference on Image Processing, Vol Ii, Proceedings*, no. 2, pp. 25–28, 2000.

[51] Z. Lin and H.-Y. Shum, "A Geometric Analysis of Light Field Rendering", *International Journal of Computer Vision*, vol. 58, no. 2, pp. 121–138, 2004.

[52] I. Ihrke, J. Restrepo, and L. Mignard-Debise, "Principles of Light Field Imaging: Briefly revisiting 25 years of research", *IEEE Signal Processing Magazine*, vol. 33, no. 5, pp. 59–69, 2016.

[53] M. Hog, N. Sabater, B. Vandame, and V. Drazic, "An Image Rendering Pipeline for Focused Plenoptic Cameras", *IEEE Transactions on Computational Imaging*, vol. 3, no. 4, pp. 811–821, 2017.

[54] M. Hog, "Light Field Editing and Rendering", Ph.D Thesis, University of Rennes 1, 2018.

[55] C. Herzog, O. de La Rochefoucauld, G. Dovillaire, X. Granier, F. Harms, X. Levecq, E. Longo, L. Mignard-Debise, and P. Zeitoun, "Comparison of reconstruction approaches for plenoptic imaging systems", in *Unconventional Optical Imaging*, C. Fournier, M. P. Georges, and G. Popescu, Eds., SPIE, May 2018, p. 104.

[56] J. N. Filipe, P. A. Assuncao, L. M. Tavora, R. Fonseca-Pinto, L. A. Thomaz, and S. M. Faria, "Improved patch-based view rendering for focused plenoptic cameras with extended depth-of-field", in *European Signal Processing Conference*, vol. 2021-Janua, 2021, pp. 680–684.

[57] D. G. Dansereau, D. L. Bongiorno, O. Pizarro, and S. B. Williams, "Light field image denoising using a linear 4D frequency-hyperfan all-in-focus filter", in *Proc. SPIE, Computational Imaging XI*, C. A. Bouman, I. Pollak, and P. J. Wolfe, Eds., vol. 8657, International Society for Optics and Photonics, Feb. 2013, 86570P.

[58] M. Alain and A. Smolic, "Light field denoising by sparse 5D transform domain collaborative filtering", *2017 IEEE 19th International Workshop on Multimedia Signal Processing, MMSP 2017*, vol. 2017-Janua, pp. 1–6, 2017.

[59] P. Allain, L. Guillo, and C. Guillemot, "Light Field Denoising Using 4D Anisotropic Diffusion", in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, May 2019, pp. 1692–1696.

[60] T. E. Bishop, S. Zanetti, and P. Favaro, "Light field superresolution", in *2009 IEEE International Conference on Computational Photography (ICCP)*, IEEE, Apr. 2009, pp. 1–9.

[61] T. Georgiev, G. Chunev, and A. Lumsdaine, "Superresolution with the focused plenoptic camera", *Computational Imaging IX*, vol. 7873, p. 78730X, 2011.

[62] 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*, vol. 34, no. 5, pp. 972–986, 2012.

[63] H. W. F. Yeung, J. Hou, X. Chen, J. Chen, Z. Chen, and Y. Y. Chung, "Light Field Spatial Super-Resolution Using Deep Spatial-Angular Interleaved CNN with Progressive Training", *IEEE Transactions on Image Processing*, vol. PP, no. c, pp. 1–13, 2018.

[64] L. Shi, H. Hassanieh, A. Davis, D. Katabi, and F. Durand, "Light Field Reconstruction Using Sparsity in the Continuous Fourier Domain", *ACM Transactions on Graphics*, vol. 34, no. 1, pp. 1–13, Dec. 2014.

[65] Y. Yoon, H.-G. Jeon, D. Yoo, J.-Y. Lee, and I. S. Kweon, "Learning a Deep Convolutional Network for Light-Field Image Super-Resolution", in *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, vol. 66, IEEE, Dec. 2015, pp. 57–65.

[66] N. K. Kalantari, T. C. Wang, and R. Ramamoorthi, "Learning-based view synthesis for light field cameras", *ACM Transactions on Graphics*, vol. 35, no. 6, pp. 1–10, 2016.

[67] G. Wu, Y. Liu, L. Fang, and T. Chai, "Spatial-Angular Attention Network for Light Field Reconstruction", vol. 14, no. 8, pp. 1–12, Jul. 2020.

[68] D. Dansereau and L. Bruton, "A 4D frequency-planar IIR filter and its application to light field processing", *Proceedings of the 2003 International Symposium on Circuits and Systems, 2003. ISCAS '03.*, vol. 4, no. January, pp. IV–476–IV–479, 2014.

[69] T. Georgiev and C. Intwala, "Light field camera design for integral view photography", *Adobe Systems Incorporated, Tech. Rep*, pp. 1–13, 2003.

[70] D. G. Dansereau, O. Pizarro, and S. B. Williams, "Linear volumetric focus for light field cameras", *ACM Transactions on Graphics*, vol. 34, no. 2, 2015.

[71] "JPEG PLENO Abstract and Executive Summary", *ISO/IEC JTC 1/SC 29/WG1 N6922*, no. February, pp. 1–5, 2015.

[72] "MPEG-I Technical Report on Immersive Media", *ISO/IEC JTC1/SC29/WG11 N17069*, no. Jully, 2017.

[73] F. Dai, J. Zhang, Y. Ma, and Y. Zhang, "Lenselet image compression scheme based on subaperture images streaming", *Proceedings - International Conference on Image Processing, ICIP*, vol. 2015-December, pp. 4733–4737, 2015.

[74] A. Vieira, H. Duarte, C. Perra, L. Tavora, and P. Assuncao, "Data formats for high efficiency coding of Lytro-Illum light fields", *5th International Conference on Image Processing, Theory, Tools and Applications (IPTA)*, pp. 494–497, 2015.

[75] A. Aggoun, "A 3D DCT Compression Algorithm For Omnidirectional Integral Images", *2006 IEEE International Conference on Acoustics Speed and Signal Processing Proceedings*, vol. 2, pp. II–517–II–520, 2006.

[76] A. Aggoun, "Compression of 3D Integral Images Using 3D Wavelet Transform", *Journal of Display Technology*, vol. 7, no. 11, pp. 586–592, Nov. 2011.

[77] C. Jia, X. Zhang, S. Wang, S. Wang, S. Pu, and S. Ma, "Light Field Image Compression Using Generative Adversarial Network Based View Synthesis", *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. PP, no. c, pp. 1–1, 2018.

[78] R. J. S. Monteiro and P. J. L. Nunes, "Light Field Image Coding using High Order Prediction Training", *Eusipco*, no. 351, pp. 1859–1863, 2018.

[79] M. Levoy, Z. Zhang, and I. McDowall, "Recording and controlling the 4D light field in a microscope", *Journal of Microscopy*, vol. 235, no. 2, pp. 144–162, 2009.

[80] L. Mignard-Debise and I. Ihrke, "Light-Field Microscopy with a Consumer Light-Field Camera", *Proceedings of the International Conference on 3D Vision (3DV)*, pp. 335–343, 2015.

[81] O. Bimber and D. Schedl, "Light-Field Microscopy: A Review", *Journal of Neurology & Neuromedicine*, vol. 4, no. 1, pp. 1–6, 2019.

[82] A. Cenedese, C. Cenedese, F. Furia, M. Marchetti, M. Moroni, and L. Shindler, "3D particle reconstruction using light field imaging", *16th Int Symp on Applications of Laser Techniques to Fluid Mechanics*, pp. 9–12, 2012.

[83] R. R. La Foy and P. Vlachos, "Multi-Camera Plenoptic Particle Image Velocimetry", *10th International Symposium on Particle Image Velocimetry*, pp. 1–17, 2013.

[84] T. W. Fahringer and B. S. Thurow, "Comparing Volumetric Reconstruction Algorithms for Plenoptic-PIV", *53rd AIAA Aerospace Sciences Meeting*, pp. 1–10, 2015.

[85] S. Shi, J. Wang, J. Ding, Z. Zhao, and T. H. New, "Parametric study on light field volumetric particle image velocimetry", *Flow Measurement and Instrumentation*, vol. 49, pp. 70–88, 2016.

[86] T. Nonn, V. Jaunet, and S. Hellman, "Spray Droplet Size and Velocity Measurement using Light-field Velocimetry", in *ICLASS 2012, 12th Triennial International Conference on Liquid Atomization and Spray Systems*, 2012, pp. 6–12.

[87] S. Hasirlioglu, M. Karthik, A. Riener, and I. Doric, "Potential of Plenoptic Cameras in the Field of Automotive Safety", in, vol. 222, Springer International Publishing, 2018, pp. 164–173.

[88] Z. Wu and Y. Liu, "Snow Removal From Light Field Images", *IEEE Access*, vol. 7, pp. 164 203–164 215, 2019.

[89] T. Yang, X. Chang, H. Su, N. Crombez, Y. Ruichek, T. Krajnik, and Z. Yan, "Raindrop Removal with Light Field Image Using Image Inpainting", *IEEE Access*, vol. 8, pp. 58 416–58 426, 2020.

[90] K. A. Skinner and M. Johnson-Roberson, "Towards real-time underwater 3D reconstruction with plenoptic cameras", in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Oct. 2016, pp. 2014–2021.

[91] A. Ghasemi and M. Vetterli, "Detecting planar surface using a light-field camera with application to distinguishing real scenes from printed photos", in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, May 2014, pp. 4588–4592.

[92] M. Feng, S. Z. Gilani, Y. Wang, and A. Mian, "3D Face Reconstruction from Light Field Images: A Model-free Approach", *ECCV*, Nov. 2018.

[93] C. Galdi, L. Younes, C. Guillemot, and J.-l. Dugelay, "A new framework for optimal facial landmark localization on light-field images", in *IEEE International Conference on Visual Communications and Image Processing*, Taichung, Taiwan, 2018, pp. 2–5.

[94] J. R. Garner, "Light Field Cameras Offer Surveillance a New Dimension", Oak Ridge National Laboratory, Tech. Rep., 2019.

[95] M. Ren, R. Liu, H. Hong, J. Ren, and G. Xiao, "Fast Object Detection in Light Field Imaging by Integrating Deep Learning with Defocusing", *Applied Sciences*, vol. 7, no. 12, p. 1309, Dec. 2017.

[96] R. Zhang, Y. Yang, W. Wang, L. Zeng, J. Chen, and S. Mcgrath, "An Algorithm for Obstacle Detection based on YOLO and Light Filed Camera", *2018 12th International Conference on Sensing Technology (ICST)*, pp. 223–226, 2018.

[97] J. Meyer, "Light Field Methods for the Visual Inspection of Transparent Objects", PhD thesis, Karlsruhe Institute of Technology, 2018.

[98] D. Tsai, D. G. Dansereau, T. Peynot, and P. Corke, "Distinguishing Refracted Features using Light Field Cameras with Application to Structure from Motion", pp. 1–8, 2018.

[99] Z. Zhou, X. Chen, and O. C. Jenkins, "LiTE: Light-field Transparency Estimation for Refractive Object Localization", 2019.

[100] Z. Zhou, T. Pan, S. Wu, H. Chang, and O. C. Jenkins, "GlassLoc: Plenoptic Grasp Pose Detection in Transparent Clutter", 2019.

[101] D. Y. P. Tsai, "Light-field features for robotic vision in the presence of refractive objects", PhD thesis, Queensland University of Technology, 2020.

[102] P. Kaveti, S. Katt, and H. Singh, "Removing Dynamic Objects for Static Scene Reconstruction using Light Fields", 2020.

[103] J. Neumann, C. Fermüller, Y. Aloimonos, and V. Brajovic, "Compound eye sensor for 3D ego motion estimation", in *International Conference on Intelligent Robots and Systems (IROS)*, vol. 4, IEEE, 2004, pp. 3712–3717.

[104] O. Johannsen, A. Sulc, and B. Goldluecke, "On Linear Structure from Motion for Light Field Cameras", in *2015 IEEE International Conference on Computer Vision (ICCV)*, IEEE, Dec. 2015, pp. 720–728.

[105] Y. Zhang, P. Yu, W. Yang, Y. Ma, and J. Yu, "Ray Space Features for Plenoptic Structure-from-Motion", *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2017-Octob, pp. 4641–4649, 2017.

[106] S. Nousias, M. Lourakis, and C. Bergeles, "Large-scale, metric structure from motion for unordered light fields", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2019-June, pp. 3287–3296, 2019.

[107] S. Nousias, M. Lourakis, P. Keane, S. Ourselin, and C. Bergeles, "A Linear Approach to Absolute Pose Estimation for Light Fields", in *2020 International Conference on 3D Vision (3DV)*, IEEE, Nov. 2020, pp. 672–681.

[108] N. Zeller, F. Quint, and U. Stilla, "Feature Based RGB-D SLAM for a Plenoptic Camera", *BW-CAR Symposium on Information and Communication Systems (SInCom)*, 2016.

[109] ——, "From the Calibration of a Light-Field Camera to Direct Plenoptic Odometry", *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 7, pp. 1004–1019, Oct. 2017.

[110] ——, "Scale-Awareness of Light Field Camera based Visual Odometry", in *European Conference on Computer Vision (ECCV)*, Munich, Germany, 2018.

[111] P. David, M. Le Pendu, and C. Guillemot, "Scene Flow Estimation From Sparse Light Fields Using a Local 4D Affine Model", *IEEE Transactions on Computational Imaging*, vol. 6, pp. 791–805, 2020.

[112] D. Tsai, D. G. Dansereau, T. Peynot, and P. I. Corke, "Image-Based Visual Servoing With Light Field Cameras", *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 912–919, 2017.

[113] H. Sardemann and H. G. Maas, "On the accuracy potential of focused plenoptic camera range determination in long distance operation", *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 114, pp. 1–9, 2016.

[114] O. Johannsen, K. Honauer, B. Goldluecke, A. Alperovich, F. Battisti, Y. Bok, M. Brizzi, M. Carli, G. Choe, M. Diebold, M. Gutsche, H.-G. Jeon, I. S. Kweon, J. Park, J. Park, H. Schilling, H. Sheng, L. Si, M. Strecke, A. Sulc, Y.-W. Tai, Q. Wang, T. C. Wang, S. Wanner, Z. Xiong, J. Yu, S. Zhang, and H. Zhu, "A Taxonomy and Evaluation of Dense Light Field Depth Estimation

Algorithms", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, vol. 2017-July, pp. 1795–1812, 2017.

[115] J. Ye and J. Yu, "Ray geometry in non-pinhole cameras: A survey", *Visual Computer*, vol. 30, no. 1, pp. 93–112, 2014.

[116] D. Brown, "Decentering Distortion of Lenses - The Prism Effect Encountered in Metric Cameras can be Overcome Through Analytic Calibration", *Photometric Engineering*, vol. 32, no. 3, pp. 444–462, 1966.

[117] A. Conrady, "Decentered Lens-Systems", *Monthly Notices of the Royal Astronomical Soceity*, vol. 79, pp. 384–390, 1919.

[118] J. Wang, F. Shi, J. Zhang, and Y. Liu, "A new calibration model of camera lens distortion", *Pattern Recognition*, vol. 41, no. 2, pp. 607–615, 2008.

[119] Z. Tang, R. G. V. Gioi, P. Monasse, Z. Tang, R. G. V. Gioi, P. Monasse, J.-m. M. A. P. Analysis, Z. Tang, R. G. V. Gioi, P. Monasse, and J.-m. Morel, "A Precision Analysis of Camera Distortion Models", *IEEE Transactions on Image Processing,*, vol. 26, no. 6, pp. 2694–2704, 2017.

[120] A. Fitzgibbon, "Simultaneous linear estimation of multiple view geometry and lens distortion", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, IEEE Comput. Soc, 2001, pp. I–125–I–132.

[121] F. Bukhari and M. N. Dailey, "Automatic radial distortion estimation from a single image", *Journal of Mathematical Imaging and Vision*, vol. 45, no. 1, pp. 31–45, 2013.

[122] O. Johannsen, C. Heinze, B. Goldluecke, and C. Perwaß, "On the calibration of focused plenoptic cameras", *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8200 LNCS, pp. 302–317, 2013.

[123] C. Heinze, "Design and test of a calibration method for the calculation of metrical range values for 3D light field cameras", Master's thesis, Hamburg University of Applied Sciences - Faculty of Engineering and Computer Science, 2014.

[124] N. Zeller, C. A. Noury, F. Quint, C. Teulière, U. Stilla, and M. Dhome, "Metric Calibration of a Focused Plenoptic Camera based on a 3D Calibration Target", *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. III-3, no. July, pp. 449–456, Jun. 2016.

[125] C. Heinze, S. Spyropoulos, S. Hussmann, and C. Perwaß, "Automated Robust Metric Calibration Algorithm for Multifocus Plenoptic Cameras", *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 5, pp. 1197–1205, 2016.

[126] N. Zeller, F. Quint, M. Sutterlin, and U. Stilla, "Investigating mathematical models for focused plenoptic cameras", in *2016 12th IEEE International Symposium on Electronics and Telecommunications (ISETC)*, IEEE, Oct. 2016, pp. 301–304.

[127] C.-a. Noury, "Etalonnage de caméra plénoptique et estimation de profondeur à partir des données brutes", Ph.D Thesis, Université Clermont Auvergne, 2019.

[128] Z. Zhang, "A flexible new technique for camera calibration", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[129] L. Alvarez, L. Gómez, and J. R. Sendra, "An algebraic approach to lens distortion by line rectification", *Journal of Mathematical Imaging and Vision*, vol. 35, no. 1, pp. 36–50, 2009.

[130] J. Mallon and P. F. Whelan, "Precise radial un-distortion of images", *Proceedings - International Conference on Pattern Recognition*, vol. 1, pp. 18–21, 2004.

[131] J. P. de Villiers, F. W. Leuschner, and R. Geldenhuys, "Centi-pixel accurate real-time inverse distortion correction", *Optomechatronic Technologies 2008*, vol. 7266, no. 1, p. 726 611, 2008.

[132] A. P. Pentland, "A New Sense for Depth of Field", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-9, no. 4, pp. 523–531, 1987.

[133] M. Subbarao, "Determining Distance from Defocused Images of Simple Objects", Tech. Rep., 1989, pp. 11 794–12 350.

[134] O. D. Faugeras, Q. T. Luong, and S. J. Maybank, "Camera self-calibration: Theory and experiments", in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1, vol. 588 LNCS, 1992, pp. 321–334.

[135] B. Triggs, "Autocalibration and the absolute quadric", in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Comput. Soc, 1997, pp. 609–614.

[136] R. Y. Tsai, "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses", *IEEE Journal on Robotics and Automation*, vol. 3, no. 4, pp. 323–344, 1987.

[137] B. K. P. Horn, "Tsai's camera calibration method revisited", vol. i, 2000.

[138] C. S. Fraser, "Digital camera self-calibration", *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 52, no. 4, pp. 149–159, 1997.

[139] T. A. Clarke and J. G. Fryer, "The development of camera calibration methods and models", *Photogrammetric Record*, vol. 16, no. 91, pp. 51–66, 1998.

[140] E. E. Hemayed, "A survey of camera self-calibration", *Proceedings - IEEE Conference on Advanced Video and Signal Based Surveillance, AVSS 2003*, pp. 351–357, 2003.

[141] F. Remondino and C. Fraser, "Digital camera calibration methods: Considerations and comparisons", in *ISPRS Commission V Symposium 'Image Engineering and Vision Metrology'*, ISPRS, 2006, pp. 266–272.

[142] M. Grossberg and S. K. Nayar, "A general imaging model and a method for finding its parameters", *Proceedings Eighth IEEE International Conference on Computer Vision*, vol. 2, pp. 108–115, 2001.

[143] R. Koch, M. Pollefeys, B. Heigl, L. V. Gool, and H. Niemann, "Calibration of hand-held camera sequences for plenoptic modeling", *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 1, 1999.

[144] V. Vaish, B. Wilburn, N. Joshi, and M. Levoy, "Using plane + parallax for calibrating dense camera arrays", in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, IEEE, 2004, pp. 2–9.

[145] T. Georgiev, A. Lumsdaine, and S. Goma, "Plenoptic Principal Planes", *Imaging and Applied Optics*, no. 2, JTuD3, 2011.

[146] D. G. Dansereau, O. Pizarro, and S. B. Williams, "Decoding, calibration and rectification for lenselet-based plenoptic cameras", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1027–1034, 2013.

[147] M. D. Grossberg and S. K. Nayar, "The raxel imaging model and ray-based calibration", *International Journal of Computer Vision*, vol. 61, no. 2, pp. 119–137, 2005.

[148] Y. Bok, H.-G. Jeon, and I. S. Kweon, "Geometric Calibration of Micro-Lens-Based Light-Field Cameras Using Line Features", in *European Conference on Computer Vision (ECCV)*, Springer International Publishing, 2014, pp. 47–61.

[149] ——, "Geometric Calibration of Micro-Lens-Based Light Field Cameras Using Line Features", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 2, pp. 287–300, 2017.

[150] C.-K. Liang and R. Ramamoorthi, "A Light Transport Framework for Lenslet Light Field Cameras", *ACM Transactions on Graphics*, vol. 34, no. 2, pp. 1–19, Mar. 2015.

[151] S. Shi, J. Ding, T. H. New, Y. Liu, and H. Zhang, "Volumetric calibration enhancements for single-camera light-field PIV", *Experiments in Fluids*, vol. 60, no. 1, p. 21, Jan. 2019.

[152] C. Hahne, A. Aggoun, V. Velisavljevic, S. Fiebig, and M. Pesch, "Baseline and Triangulation Geometry in a Standard Plenoptic Camera", *International Journal of Computer Vision*, vol. 126, no. 1, pp. 21–35, 2018.

[153] P. Zhou, W. Cai, Y. Yu, Y. Zhang, and G. Zhou, "A two-step calibration method of lenslet-based light field cameras", *Optics and Lasers in Engineering*, vol. 115, pp. 190–196, 2019.

[154] F. Bergamasco, A. Albarelli, L. Cosmo, A. Torsello, E. Rodola, and D. Cremers, "Adopting an unconstrained ray model in light-field cameras for 3D shape reconstruction", in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2015, pp. 3003–3012.

[155] F. Bergamasco, A. Albarelli, E. Rodola, and A. Torsello, "Can a fully unconstrained imaging model be applied effectively to central cameras?", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1391–1398, 2013.

[156] E. Lilienblum and B. Michaelis, "Optical 3D surface reconstruction by a multi-period phase shift method", *Journal of Computers (Finland)*, vol. 2, no. 2, pp. 73–83, 2007.

[157] E. M. Hall, T. W. Fahringer, D. R. Guildenbecher, and B. S. Thurow, "Volumetric calibration of a plenoptic camera", *Applied Optics*, vol. 57, no. 4, p. 914, 2018.

[158] K. H. Strobl and M. Lingenauber, "Stepwise calibration of focused plenoptic cameras", *Computer Vision and Image Understanding*, vol. 145, pp. 140–147, 2016.

[159] N. Zeller, F. Quint, and U. Stilla, "Calibration and accuracy analysis of a focused plenoptic camera", *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. II-3, no. September, pp. 205–212, 2014.

[160] S. O'Brien, J. Trumpf, V. Ila, and R. Mahony, "Calibrating light-field cameras using plenoptic disc features", in *2018 International Conference on 3D Vision (3DV)*, IEEE, 2018, pp. 286–294.

[161] S. Nousias, F. Chadebecq, J. Pichat, P. Keane, S. Ourselin, and C. Bergeles, "Corner-Based Geometric Calibration of Multi-focus Plenoptic Cameras", *Proceedings of the IEEE International Conference on Computer Vision*, pp. 957–965, 2017.

[162] C. Zhang, Z. Ji, and Q. Wang, "Decoding and calibration method on focused plenoptic camera", *Computational Visual Media*, vol. 2, no. 1, pp. 57–69, 2016.

[163] ——, "Unconstrained Two-parallel-plane Model for Focused Plenoptic Cameras Calibration", pp. 1–20, 2016.

[164] 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.

[165] J. Sun, C. Xu, B. Zhang, S. Wang, M. M. Hossain, H. Qi, and H. Tan, "Geometric calibration of focused light field camera for 3-D flame temperature measurement", in *Conference Record - IEEE Instrumentation and Measurement Technology Conference*, Jul. 2016.

[166] C. A. Noury, C. Teulière, and M. Dhome, "Light-Field Camera Calibration from Raw Images", *DICTA 2017 – International Conference on Digital Image Computing: Techniques and Applications*, pp. 1–8, 2017.

[167] D. Nister, "An efficient solution to the five-point relative pose problem", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 756–770, Jun. 2004.

[168] V. Lepetit, F. Moreno-Noguer, and P. Fua, "EPnP: An accurate O(n) solution to the PnP problem", *International Journal of Computer Vision*, vol. 81, no. 2, pp. 155–166, 2009.

[169] Y. Wang, J. Qiu, C. Liu, D. He, X. Kang, J. Li, and L. Shi, "Virtual Image Points Based Geometrical Parameters' Calibration for Focused Light Field Camera", *IEEE Access*, vol. 6, no. c, pp. 71 317–71 326, 2018.

[170] 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", *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3280–3287, 2013.

[171] C. M. Thomason, B. S. Thurow, and T. W. Fahringer, "Calibration of a Microlens Array for a Plenoptic Camera", *52nd Aerospace Sciences Meeting*, no. January, pp. 1–18, 2014.

[172] S. Xu, Z. L. Zhou, and N. Devaney, "Multi-view image restoration from plenoptic raw images", *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9009, pp. 3–15, 2015.

[173] P. Suliga and T. Wrona, "Microlens array calibration method for a light field camera", *Proceedings of the 19th International Carpathian Control Conference (ICCC)*, pp. 19–22, 2018.

[174] S. Li, Y. Zhu, C. Zhang, Y. Yuan, and H. Tan, "Rectification of images distorted by microlens array errors in plenoptic cameras", *Sensors (Switzerland)*, vol. 18, no. 7, 2018.

[175] L. Mignard-Debise, J. Restrepo, and I. Ihrke, "A Unifying First-Order Model for Light-Field Cameras: The Equivalent Camera Array", *IEEE Transactions on Computational Imaging*, vol. 3, no. 4, pp. 798–810, 2017.

[176] T. Scheimpflug, "Improved Method and Apparatus for the Systematic Alteration or Distortion of Plane Pictures and Images by Means of Lenses and Mirrors for Photograpy and for other purposes", GB-190401196A, 1904.

[177] M. Subbarao and G. Surya, "Depth from defocus: A spatial domain approach", *International Journal of Computer Vision*, vol. 13, no. 3, pp. 271–294, 1994.

[178] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", in *KDD*, 1996.

[179] L. Kneip, D. Scaramuzza, and R. Siegwart, "A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2969–2976, 2011.

[180] J. Ens and P. Lawrence, "An Investigation of Methods for Determining Depth from Focus", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 2, pp. 97–108, 1993.

[181] F. Mannan and M. S. Langer, "Blur calibration for depth from defocus", in *13th Conference on Computer and Robot Vision (CRV)*, 2016, pp. 281–288.

[182] M. Subbarao, "Parallel Depth Recovery By Changing Camera Parameters", pp. 149–155, 1988.

[183] F. Mannan and M. S. Langer, "What is a good model for depth from defocus?", in *13th Conference on Computer and Robot Vision (CRV)*, 2016, pp. 273–280.

[184] C. H. Chen, H. Zhou, and T. Ahonen, "Blur-aware disparity estimation from defocus stereo images", *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015 Inter, pp. 855–863, 2015.

[185] O. Fleischmann and R. Koch, "Lens-Based Depth Estimation for Multi-focus Plenoptic Cameras", in *Electronics and Power*, 9, vol. 24, 2014, pp. 410–420.

[186] R. Ferreira and N. Goncalves, "Fast and accurate micro lenses depth maps for multi-focus light field cameras", *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9796 LNCS, pp. 309–319, 2016.

[187] L. Palmieri and R. Koch, "Optimizing the Lens Selection Process for Multi-focus Plenoptic Cameras and Numerical Evaluation", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, vol. 2017-July, pp. 1763–1774, 2017.

[188] M. W. Tao, S. Hadap, J. Malik, and R. Ramamoorthi, "Depth from combining defocus and correspondence using light-field cameras", *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, vol. 2, pp. 673–680, 2013.

[189] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms", *International Journal of Computer Vision*, vol. 47, no. 1-3, pp. 7–42, 2002.

[190] B. Zitová and J. Flusser, "Image registration methods: A survey", *Image and Vision Computing*, vol. 21, no. 11, pp. 977–1000, 2003.

[191] J. Li and N. M. Allinson, "A comprehensive review of current local features for computer vision", *Neurocomputing*, vol. 71, no. 10-12, pp. 1771–1787, 2008.

[192] S. A. K. Tareen and Z. Saleem, "A comparative analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK", *2018 International Conference on Computing, Mathematics and Engineering Technologies: Invent, Innovate and Integrate for Socioeconomic Development, iCoMET 2018 - Proceedings*, vol. 2018-Janua, pp. 1–10, 2018.

[193] J. Konz, N. Zeller, F. Quint, and U. Stilla, "Depth Estimation from Micro Images of a Plenoptic Camera", in *BW-CAR Symposium on Information and Communication Systems (SInCom)*, 2016, pp. 17–23.

[194] T. Kanade, A. Yoshida, K. Oda, H. Kano, and M. Tanaka, "Stereo machine for video-rate dense depth mapping and its new applications", in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1996, pp. 196–202.

[195] M. J. Hannah, "Computer Matching of Areas in Stereo Images", PhD thesis, Stanford University, 1974, p. 134.

[196] S. Birchfield and C. Tomasi, "A pixel dissimilarity measure that is insensitive to image sampling", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 4, pp. 401–406, 1998.

[197] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence", *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 801 LNCS, pp. 151–158, 1994.

[198] Williem and I. K. Park, "Cost aggregation benchmark for light field depth estimation", *Journal of Visual Communication and Image Representation*, vol. 56, pp. 38–51, 2018.

[199] P. Grossmann, "Depth from focus", *Pattern Recognition Letters*, vol. 5, no. 1, pp. 63–69, 1987.

[200] S. H. Lai, C. W. Fu, and S. Chang, "A Generalized Depth Estimation Algorithm with a Single Image", vol. 14, no. 4, pp. 405–411, 1992.

[201] F. Perez Nava and J. P. Luke, "Simultaneous estimation of super-resolved depth and all-in-focus images from a plenoptic camera", in *2009 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video*, IEEE, May 2009, pp. 1–4.

[202] M.-J. Kim, T.-H. Oh, and I. S. Kweon, "Cost-aware depth map estimation for Lytro camera", in *2014 IEEE International Conference on Image Processing (ICIP)*, IEEE, Oct. 2014, pp. 36–40.

[203] H.-G. Jeon, J. Park, G. Choe, J. Park, Y. Bok, Y.-W. Tai, and I. S. Kweon, "Accurate depth map estimation from a lenslet light field camera", in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2015, pp. 1547–1555.

[204] H.-g. Jeon, J. Park, G. Choe, J. Park, Y. Bok, Y.-W. Tai, and I. S. Kweon, "Depth from a Light Field Image with Learning-based Matching Costs", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8828, no. c, pp. 1–1, 2018.

[205] J. Peng, Z. Xiong, Y. Zhang, D. Liu, and F. Wu, "LF-fusion: Dense and accurate 3D reconstruction from light field images", in *2017 IEEE Visual Communications and Image Processing (VCIP)*, IEEE, Dec. 2017, pp. 1–4.

[206] Y. Zhang, W. Dai, M. Xu, J. Zou, X. Zhang, and H. Xiong, "Depth Estimation from Light Field Using Graph-Based Structure-Aware Analysis", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. PP, no. c, pp. 1–1, 2019.

[207] Y. Anisimov, O. Wasenmuller, and D. Stricker, "Rapid Light Field Depth Estimation with Semi-Global Matching", in *2019 IEEE 15th International Conference on Intelligent Computer Communication and Processing (ICCP)*, IEEE, Sep. 2019, pp. 109–116.

[208] D. G. Dansereau, B. Girod, and G. Wetzstein, "LiFF: Light Field Features in Scale and Depth", in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2019, pp. 8034–8043.

[209] M. Alain and A. Smolic, "A Spatio-Angular Binary Descriptor for Fast Light Field Inter View Matching", in *Proceedings - International Conference on Image Processing, ICIP*, vol. 2020-Octob, 2020, pp. 2636–2640.

[210] Z. Xiao, M. E. Zhang, H. Jin, and C. Guillemot, "A light field FDL-HSIFT feature in scale-disparity space", in *IEEE International Conference on Image Processing (ICIP)*, IEEE, Ed., 2021.

[211] S. Wanner, J. Fehr, and B. Jähne, "Generating EPI representations of 4D light fields with a single lens focused plenoptic camera", *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6938 LNCS, no. PART 1, pp. 90–101, 2011.

[212] S. Wanner and B. Goldluecke, "Globally consistent depth labeling of 4D light fields", in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2012, pp. 41–48.

[213] Z. Yu, X. Guo, H. Ling, A. Lumsdaine, and J. Yu, "Line assisted light field triangulation and stereo matching", in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 2792–2799.

[214] I. Tosic and K. Berkner, "Light Field Scale-Depth Space Transform for Dense Depth Estimation", in *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, IEEE, Jun. 2014, pp. 441–448.

[215] S. Xu, "4D Light Field Reconstruction and Scene Depth Estimation from Plenoptic Camera Raw Images", Ph.D Thesis, National University of Ireland, Galway, 2016.

[216] C. Chen, H. Lin, Z. Yu, S. B. Kang, and J. Yu, "Light Field Stereo Matching Using Bilateral Statistics of Surface Cameras", in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Jun. 2014, pp. 1518–1525.

[217] S. Zhang, H. Sheng, C. Li, J. Zhang, and Z. Xiong, "Robust depth estimation for light field via spinning parallelogram operator", *Computer Vision and Image Understanding*, vol. 145, pp. 148–159, Apr. 2016.

[218] H. Sheng, P. Zhao, S. Zhang, J. Zhang, and D. Yang, "Occlusion-aware depth estimation for light field using multi-orientation EPIs", *Pattern Recognition*, vol. 74, pp. 587–599, 2018.

[219] H. Ma, H. Li, Z. Qian, S. Shi, and T. Mu, "VommaNet: an End-to-End Network for Disparity Estimation from Reflective and Texture-less Light Field Images", 2018.

[220] C. Shin, H.-G. Jeon, Y. Yoon, I. S. Kweon, and S. J. Kim, "EPINET: A Fully-Convolutional Neural Network Using Epipolar Geometry for Depth from Light Field Images", 2018.

[221] X. Liu, D. Fu, C. Wu, and Z. Si, "The Depth Estimation Method Based on Double-Cues Fusion for Light Field Images", in *Proceedings of the 11th International Conference on Modelling, Identification and Control (ICMIC2019), Lecture Notes in Electrical Engineering 582*, 2020, pp. 719–726.

[222] J. Jin and J. Hou, "Occlusion-aware Unsupervised Learning of Depth from 4-D Light Fields", pp. 1–10, Jun. 2021.

[223] T. Leistner, H. Schilling, R. Mackowiak, S. Gumhold, and C. Rother, "Learning to Think Outside the Box: Wide-Baseline Light Field Depth Estimation with EPI-Shift", no. c, 2019.

[224] O. Johannsen, A. Sulc, and B. Goldluecke, "What Sparse Light Field Coding Reveals about Scene Structure", *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3262–3270, 2016.

[225] S. Heber, W. Yu, and T. Pock, "U-shaped Networks for Shape from Light Field", *Procedings of the British Machine Vision Conference 2016*, vol. 1, no. 1, pp. 37.1–37.12, 2016.

[226] K. Li, J. Zhang, R. Sun, X. Zhang, and J. Gao, "EPI-based Oriented Relation Networks for Light Field Depth Estimation", pp. 1–11, 2020.

[227] C. T. Huang, *Empirical Bayesian Light-Field Stereo Matching by Robust Pseudo Random Field Modeling*, 2018.

[228] J. P. B. L. Custodio, "Depth Estimation using Light-Field Cameras", Master Thesis, Universidade de Coimbra, 2014.

[229] F. Cunha, L. A. Thomaz, L. M. N. Tavora, P. A. A. Assuncao, R. Fonseca-Pinto, and S. M. M. Faria, "Robust Depth Estimation From Multi-Focus Plenoptic Images", in *2020 IEEE International Conference on Image Processing (ICIP)*, IEEE, Oct. 2020, pp. 2626–2630.

[230] N. Zeller, F. Quint, and U. Stilla, "Depth estimation and camera calibration of a focused plenoptic camera for visual odometry", *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 118, pp. 83–100, 2016.

[231] R. T. Held, E. A. Cooper, and M. S. Banks, "Blur and disparity are complementary cues to depth", *Current Biology*, vol. 22, no. 5, pp. 426–431, 2012.

[232] Y. Y. Schechner and N. Kiryati, "Depth from Defocus vs. stereo: How different really are they?", *International Journal of Computer Vision*, vol. 39, no. 2, pp. 141–162, 2000.

[233] V. Vaish, R. Szeliski, C. L. Zitnick, S. B. Kang, and M. Levoy, "Reconstructing occluded surfaces using synthetic apertures: Stereo, focus and robust measures", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 2331–2338, 2006.

[234] T. Xian and M. Subbarao, "Depth-from-defocus: blur equalization technique", *Two- and Three-Dimensional Methods for Inspection and Metrology IV*, vol. 6382, 63820E, 2006.

[235] J. Kiefer, "Sequential Minimax Search for a Maximum", *Proceedings of the American Mathematical Society*, vol. 4, no. 3, p. 502, 1953.

[236] P. J. Besl and N. D. McKay, "A Method for Registration of 3D Shapes", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.

[237] Y. Chen and G. Medioni, "Object modeling by registration of multiple range images", in *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, 1992, pp. 2724–2729.

[238] N. Zeller, F. Quint, and U. Stilla, "Narrow Field-of-View Visual Odometry Based on a Focused Plenoptic Camera", *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. II-3/W4, no. March, pp. 285–292, 2015.

[239] D. G. Dansereau, G. Schuster, J. Ford, and G. Wetzstein, "A Wide-Field-of-View Monocentric Light Field Camera", in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jul. 2017, pp. 3757–3766.

[240] G. M. Schuster, D. G. Dansereau, G. Wetzstein, and J. E. Ford, "Panoramic single-aperture multi-sensor light field camera", *Optics Express*, vol. 27, no. 26, p. 37 257, Dec. 2019.

[241] P. Trouvé, "Conception conjointe optique/traitement pour un imageur compact à capacité 3D", PhD thesis, Ecole Centrale de Nantes (ECN), 2012.

[242] V. Sitzmann, S. Diamond, Y. Peng, X. Dun, S. Boyd, W. Heidrich, F. Heide, and G. Wetzstein, "End-to-end Optimization of Optics and Image Processing for Achromatic Extended Depth of Field and Super-resolution Imaging", *ACM Trans. Graph. (SIGGRAPH)*, 2018.