HAL
open science

Camera Models and algorithms for 3D video content creation

Sergi Pujades Rocamora

To cite this version:

Sergi Pujades Rocamora. Camera Models and algorithms for 3D video content creation. Image Processing [eess.IV]. Université Grenoble Alpes, 2015. English. NNT : 2015GREAM039 . tel-01281363

HAL Id: tel-01281363

https://theses.hal.science/tel-01281363v1

Submitted on 2 Mar 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Mathématiques et Informatique**

Bourse: Action 3DS

Présentée par

Sergi PUJADES ROCAMORA

Thèse dirigée par **Rémi RONFARD**
et codirigée par **Frédéric DEVERNAY**

préparée au sein **du Laboratoire d'Informatique de Grenoble á l'INRIA Rhône-Alpes**
et de **l'Ecole Doctorale de Mathématiques, Sciences et Technologies de l'Information**

Modèles de caméras et algorithmes pour la création de contenu video 3D

Thèse soutenue publiquement le **14 octobre 2015**,
devant le jury composé de :

M. James CROWLEY

Professor at Grenoble INP, France, Président

Ms. Luce MORIN

Professor at INSA Rennes, France, Rapporteur

M. Jean-Yves GUILLEMAUT

Assistant Professor at University of Surrey, United Kingdom, Rapporteur

M. George DRETTAKIS

Research Director at INRIA Sophia-Antipolis, France, Examinateur

M. Aljoscha SMOLIC

Senior Research Scientist at Disney Research Zurich, Switzerland, Examinateur

M. Rémi RONFARD

Researcher at INRIA Grenoble, France, Examinateur

M. Frédéric DEVERNAY

Researcher at INRIA Grenoble, France, Examinateur

For you, dear reader.

Abstract

Optics with long focal length have been extensively used for shooting 2D cinema and television, either to virtually get closer to the scene or to produce an aesthetical effect through the deformation of the perspective. However, in 3D cinema or television, the use of long focal length either creates a "cardboard effect" or causes visual divergence. To overcome this problem, state-of-the-art methods use disparity mapping techniques, which is a generalization of view interpolation, and generate new stereoscopic pairs from the two image sequences. We propose to use more than two cameras to solve for the remaining issues in disparity mapping methods.

In the first part of the thesis, we review the causes of visual fatigue and visual discomfort when viewing a stereoscopic film. We then model the depth perception from stereopsis of a 3D scene shot with two cameras, and projected in a movie theater or on a 3DTV. We mathematically characterize this 3D distortion, and derive the mathematical constraints associated with the causes of visual fatigue and discomfort. We illustrate these 3D distortions with a new interactive software, "The Virtual Projection Room".

In order to generate the desired stereoscopic images, we propose to use image-based rendering. These techniques usually proceed in two stages. First, the input images are warped into the target view, and then the warped images are blended together. The warps are usually computed with the help of a geometric proxy (either implicit or explicit). Image blending has been extensively addressed in the literature and a few heuristics have proven to achieve very good performance. Yet the combination of the heuristics is not straightforward, and requires manual adjustment of many parameters.

In this thesis, we propose a new Bayesian approach to the problem of novel view synthesis, based on a generative model taking into account the uncertainty of the image warps in the image formation model. The Bayesian formalism allows us to deduce the energy of the generative model and to compute the desired images as the Maximum a Posteriori estimate. The method outperforms state-of-the-art image-based rendering techniques on challenging datasets. Moreover, the energy equations provide a formalization of the heuristics widely used in image-based rendering techniques. Besides, the proposed generative model also addresses the problem of super-resolution, allowing to render images at a higher resolution than the initial ones.

In the last part of this thesis, we apply the new rendering technique to the case of the stereoscopic zoom and show its performance.

Keywords image-based-rendering, geometric uncertainty, Bayesian approach, stereoscopic cinematography, 3DTV.

Résumé

Des optiques à longue focale ont été souvent utilisées dans le cinéma 2D et la télévision, soit dans le but de se rapprocher de la scène, soit dans le but de produire un effet esthétique grâce à la déformation de la perspective. Toutefois, dans le cinéma ou la télévision 3D, l'utilisation de longues focales crée le plus souvent un "effet carton" ou de la divergence oculaire. Pour résoudre ce problème, les méthodes de l'état de l'art utilisent des techniques de transformation de la disparité, qui sont une généralisation de l'interpolation de points de vue. Elles génèrent de nouvelles paires stéréoscopiques à partir des deux séquences d'images originales. Nous proposons d'utiliser plus de deux caméras pour résoudre les problèmes non résolus par les méthodes de transformation de la disparité.

Dans la première partie de la thèse, nous passons en revue les causes de la fatigue visuelle et de l'inconfort visuel lors de la visualisation d'un film stéréoscopique. Nous modélisons alors la perception de la profondeur de la vision stéréoscopique d'une scène filmée en 3D avec deux caméras, et projetée dans une salle de cinéma ou sur un téléviseur 3D. Nous caractérisons mathématiquement cette distorsion 3D, et formulons les contraintes mathématiques associées aux causes de la fatigue visuelle et de l'inconfort. Nous illustrons ces distorsions 3D avec un nouveau logiciel interactif, la "salle de projection virtuelle".

Afin de générer les images stéréoscopiques souhaitées, nous proposons d'utiliser le rendu basé image. Ces techniques comportent généralement deux étapes. Tout d'abord, les images d'entrée sont transformées vers la vue cible, puis les images transformées sont mélangées. Les transformations sont généralement calculées à l'aide d'une géométrie intermédiaire (implicite ou explicite). Le mélange d'images a été largement étudié dans la littérature et quelques heuristiques permettent d'obtenir de très bonnes performances. Cependant, la combinaison des heuristiques proposées n'est pas simple et nécessite du réglage manuel de nombreux paramètres.

Dans cette thèse, nous proposons une nouvelle approche bayésienne au problème de synthèse de nouveaux points de vue. Le modèle génératif proposé tient compte de l'incertitude sur la transformation d'image. Le formalisme bayésien nous permet de déduire l'énergie du modèle génératif et de calculer les images désirées correspondant au maximum a posteriori. La méthode dépasse en termes de qualité les techniques de l'état de l'art du rendu basé image sur des jeux de données complexes. D'autre part, les équations de l'énergie fournissent une formalisation des heuristiques largement utilisés dans les techniques de rendu basé image. Le modèle génératif proposé aborde également le problème de la super-résolution, permettant de rendre des images à une résolution plus élevée que les images de départ.

Dans la dernière partie de cette thèse, nous appliquons la nouvelle technique de rendu au cas du zoom stéréoscopique et nous montrons ses performances.

Mots-Clés rendu basé image, incertitude geometrique, formalisme bayesien, cinmatographie stereoscopique, TV3D.

Publications related with this thesis

1. **S. Pujades**, F. Devernay and B. Goldluecke, "Bayesian View Synthesis and Images-Based Rendering Principles", in *Conference on Computer Vision and Pattern Recognition (CVPR)*, Columbus (USA), Jun. 2014.

2. **S. Pujades**, F. Devernay, "Viewpoint Interpolation: Direct and Variational Methods", in *International Conference on Image Processing (ICIP)*, Paris (France), Oct. 2014.

3. **S. Pujades**, L. Boiron, R. Ronfard, F. Devernay "Dynamic Stereoscopic Previz", in *International Conference on 3D Imaging (IC3D)*, Liège (Belgium), Dec. 2014.

4. **S. Pujades**, F. Devernay "System for generating an optical illusion in binocular vision and associated method", *WO2015028626A1* Filing date: 2nd Sep. 2013.

Acknowledgments

This work would not have been possible without the help of a lot of people. I would like to thank them all.

Un grand merci à la Caisse des Dépôts et Consignations qui a financé le projet Action 3DS. Sans cette aide, ces travaux n'auraient pas été possibles.

Merci beaucoup à vous, Rémi et Fred, de m'avoir donné la possibilité de faire cette thèse. L'experience a été intense et m'a beaucoup enrichi. Merci de m'avoir encadré!

Thank you, Luce Morin and Jean-Yves Guillemaut for reviewing the manuscript and helping me improve it. And thank you, Aljoscha Smolic and Georges Drettakis for examining my work. I enjoyed your questions and feedback, which have already sparked more ideas!

Bastian, Dir möchte ich auch danken. Deine Arbeit und Vertrauen mit cocolib hat mir viel geholfen. Vielen Dank!

Thank you Jim for your advice, which guided me during the journey. Et merci aussi à toi Catherine, pour ton efficacité et ta bonne humeur. Tu m'as beaucoup aidé à parcourir les chemins ténébreux de la bureaucratie.

Yves, je te remercie de m'avoir tant appris sur la stéréoscopie et la cinématographie pendant mon expérience à Binocle3D.

Je remercie toute l'équipe du tournage "Endless Night" pour vos belles images et vos retours.

Laurent, je tiens à te remercier très spécialement. D'un côté, pour tous les scripts et modèles Blender que tu as créés, de l'autre, pour nos échanges au jour le jour. Ça a été un vrai plaisir de travailler et voyager avec toi.

A ti Julian también te quiero agradecer el tiempo que pasamos juntos. Ha sido un placer conocerte y poder compartir trabajo, alegrías, sufrimiento y ocio contigo.

Greg, merci de m'avoir aidé avec ces interminables fichiers de config et ces scripts. Alexandre, merci pour ta bonne humeur dans le bureau.

A tu Pau, no se si agraïr-te o maleïr-te per haver-me introduit al món Bayesià! Va ser un plaer treballar amb tu quan feies la tèsi i totes aquelles converses als sofas de l'Inria m'han ajudat molt durant la meva tèsi. Crec que un cop acabada la tèsi, m'inclino per agrair-t'ho! He après molt amb tu. Gràcies Pau!

Thierry, je voulais aussi te remercier pour ton point de vue critique et constructif, ainsi que pour ces pauses café, qui sans doute me manqueront. Merci!

Jean et Cathy, je tiens aussi à vous remercier pour tout le soutien que vous m'avez apporté au long de ces dernières années, et très spécialement cet été à Agon. Merci à vous!

A vosaltres, David, Laura, Geralyn i Victor, també us volia agraïr la vostra paciència i ànims. Sobretot, per aquest estiu que s'ha escapat com la sorra de la platja entre els dits. Specially I wanted to thank you Geralyn and David for your support in my last summer sprint in LA. Your helped me get through it!

A vosaltres, Lluís i Mercè, moltes gràcies pel votre suport incondicional. Sempre heu estat al meu costat, fins i tot quan ereu a l'altra banda del món.

A tu Bruna també et vull donar les gràcies per totes les vegades que em vas venir a buscar al despatx dient que ja estava bé de treballar i que ja era hora d'anar a jugar. Som-hi!

I a tu Céline, per ser una companya de viatge tan fantàstica! Moltes gràcies de tot cor!

Sergi Pujades Rocamora
Grenoble, November 11, 2015

Notation

In an effort to provide a uniform notation with other computer vision references, in this thesis we use the notation of the book *Computer Vision - Algorithms and Applications* (Szeliski, 2010). To introduce the notation we reproduce its Section 1.5: *A note on notation.*

"For better or worse, the notation found in computer vision and multi-view geometry textbooks tends to vary all over the map (Faugeras, 1993; Hartley and Zisserman, 2004; Girod *et al.*, 2000; Faugeras and Luong, 2004; Forsyth and Ponce, 2002). In this book, I use the convention I first learned in my high school physics class (and later multi-variate calculus and computer graphics courses), which is that vectors $\boldsymbol{v}$ are lower case bold, matrices $\boldsymbol{M}$ are upper case bold, and scalars (T, s) are mixed case italic. Unless otherwise noted, vectors operate as column vectors, i.e., they post-multiply matrices, $\boldsymbol{M}\boldsymbol{v}$, although they are sometimes written as comma-separated parenthesized lists $\boldsymbol{x} = (x, y)$ instead of bracketed column vectors $\boldsymbol{x} = [x\ y]^\top$. Some commonly used matrices are $\boldsymbol{R}$ for rotations, $\boldsymbol{K}$ for calibration matrices, and $\boldsymbol{I}$ for the identity matrix. Homogeneous coordinates are denoted with a tilde over the vector, e.g. $\tilde{\boldsymbol{x}} = (\tilde{x}, \tilde{y}, \tilde{w}) = \tilde{w}(x, y, 1) = \tilde{w}\bar{\boldsymbol{x}}$ in $\mathcal{P}^2$. The cross product operator in matrix form is denoted by $[\ ]_\times$."

Richard Szelisky, 2010.

In addition we introduce the following element notation for the components of vectors and matrices. The coordinates of a vector $\boldsymbol{x}$ are notated with sub-indices: $\boldsymbol{x} = (\boldsymbol{x}_x, \boldsymbol{x}_y, \boldsymbol{x}_z, \boldsymbol{x}_w)$. In the case where $\boldsymbol{x}$ has already a sub-index, e.g. $\boldsymbol{x}_i$, we use the accolades to enumerate the components: $\boldsymbol{x}_i = (\boldsymbol{x}_i[1], \boldsymbol{x}_i[2], \boldsymbol{x}_i[3], \boldsymbol{x}_i[4])$. For a matrix $\boldsymbol{M}$, the first sub-index denotes the row and second sub-index the column, thus $\boldsymbol{M}_{xx}$ is the element in the first row and column. For generic size matrices we use the accolades notation $\boldsymbol{M}[i, j]$ to denote the element on the i'th row and j'th column.

Next we provide a table of used symbols for quick reference:

$\mathbb{R}$	set of real numbers
$\lvert a\rvert = \sqrt{a^2}$	absolute value, $a \in \mathbb{R}$
$\mathbf{x} = (x, y)$	2D image point
$\bar{\mathbf{x}} = (x, y, 1)$	3D extended coordinates
$\tilde{\mathbf{x}} = \tilde{w}(x, y, 1)$	3D homogeneous coordinates
$\mathbf{P}$	3×4 camera projection matrix
$\mathbf{K}$	3×3 matrix with the camera intrinsic parameters
$\mathbf{R}$	3×3 rotation matrix
$\mathbf{t}$	3D translation vector
b	baseline (or interaxial) between cameras
H	convergence window distance
W	convergence window width
b'	spectator interocular distance
H'	screen to spectator distance
W'	screen width
f	focal length (in pixels units unless specified otherwise)
d	disparity (in pixels units unless specified otherwise)
w	width of the image in pixels
Ω_i	input image domain
Γ	target image domain
$\tau_i : \Omega_i \to \Gamma$	backward warp map from input image to target image
$\beta_i : \Gamma \to \Omega_i$	forward warp map from target image to input image
$m_i : \Omega_i \to \{0, 1\}$	visibility map of the input image
$V_i \in \Omega_i$	set of the visible elements in Ω_i
ε_s	sensor noise error
ε_g	image noise error due to geometric uncertainty
σ_s^2	sensor noise variance
σ_z^2	variance of a depth estimate in geometric units
σ_g^2	variance of an intensity measure due to geometric uncertainty
σ_n^2	variance of a depth estimate along the surface's normal vector
$\phi : \mathbb{R} \to \mathbb{R}$	disparity mapping function
$\Phi : \mathbb{R} \to \mathbb{R}$	depth mapping function
$\mathbf{\Phi} : \mathbb{R}^3 \to \mathbb{R}^3$	world distortion function

Contents

1 Introduction **1**

1.1 Motivation 1

1.2 The Research Problem 2

1.3 Contributions 3

1.4 Thesis Outline 5

2 Depth Perception and Visual Fatigue **7**

2.1 Depth Cues 7

2.1.1 Monoscopic Cues 7

2.1.2 Stereoscopic Depth Cues 8

2.1.3 Conflicting Depth Cues 9

2.1.4 Inconsistent Depth Cues 11

2.2 Visual Comfort and Visual Fatigue 12

2.2.1 Vergence-Accomodation Conflict 14

2.2.2 Horizontal Disparity Limits 14

2.3 Summary 16

3 Stereoscopic Filming: a Geometric Study **17**

3.1 3D Transformations and Camera Matrices 17

3.1.1 3D Translations and Rotations 18

3.1.2 Perspective 3D to 2D Projection 18

3.1.3 Pinhole Camera Model 19

3.1.4 Epipolar Geometry Between Two Cameras 20

3.1.5 Two Rectified Cameras 21

3.1.6 The Disparity 22
3.1.7 3D to 3D Transformations: the Reconstruction Matrix 24
3.2 Stereoscopic Filming: Acquisition and Projection 26
3.2.1 Perceived Depth from Stereopsis 26
3.2.2 Perceived Position from Stereopsis 28
3.2.3 Ocular Divergence Limits 30
3.2.4 Roundness Factor 32
3.2.5 Relative Perceived Size of Objects 34
3.2.6 Changing the Projection Geometry 35
3.2.7 The Ideal Viewing Distance 36
3.3 The Virtual Projection Room 37
3.4 Adapting the Content to the Width of the Screen 40
3.4.1 Modifying the Perceived Depth 44
3.4.2 Disparity Mapping Functions 47
3.5 Filming with Long Focal Lengths: Ocular Divergence vs. Roundness 49
3.5.1 Limitations of the State of the Art 50
3.5.2 Why Do Artists Use Long Focal Lengths? 51
3.5.3 Proposed Solutions 52
3.5.4 Research Questions 54

4 Bayesian Modeling of Image-Based Rendering 55
4.1 Motivation 55
4.2 Related Work 58
4.2.1 Image-Based Rendering 58
4.2.2 3D Reconstruction Methods 64
4.3 Formalizing Unstructured Lumigraph 67
4.3.1 The Bayesian Formalism 67
4.3.2 Novel View Synthesis Generative Model 68
4.4 Simplified Camera Configuration Experiments 79
4.4.1 Structured Light Field Datasets 80
4.4.2 Numerical Evaluation 80
4.4.3 Processing Time 82
4.5 Experiments on Generic Camera Configuration 84
4.5.1 Input Generation: 3D Reconstruction and Uncertainty Computation 84
4.5.2 Unstructured View Synthesis Model 94
4.5.3 Datasets 100
4.5.4 Numerical Evaluation 100

4.5.5 Processing Time . . . 102
4.5.6 Generic Configuration Results . . . 102
4.5.7 Discussion and Hints for Improvement . . . 103
4.6 Relation to the Principles of IBR . . . 108
4.6.1 Use of Geometric Proxies & Unstructured Input . . . 108
4.6.2 Epipole Consistency . . . 109
4.6.3 Minimal Angular Deviation . . . 109
4.6.4 Resolution Sensitivity . . . 110
4.6.5 Equivalent Ray Consistency . . . 110
4.6.6 Continuity . . . 111
4.6.7 Real-Time . . . 112
4.6.8 Balance Between Properties . . . 112
4.7 Summary of Contributions . . . 112

5 The Stereoscopic Zoom 115
5.1 Being On the Field! . . . 116
5.1.1 The Mise-en-Scene . . . 116
5.1.2 The Quadri-Rig . . . 118
5.1.3 Proof of Concept . . . 129
5.1.4 Quadri-Rig Discussion . . . 130
5.2 Distort the World! . . . 136
5.2.1 The Mise-en-Scene . . . 136
5.2.2 The Multi-Rig . . . 137
5.2.3 Proof of Concept . . . 146
5.2.4 Tri-Rig Discussion . . . 149
5.3 Discussion and Conclusion . . . 150
5.3.1 Tri-Rig vs. Quadri-Rig . . . 150
5.3.2 Actual Implementation . . . 151
5.3.3 Autonomous Calibration and Depth Computation . . . 151
5.3.4 Future Evaluation . . . 152

6 Conclusion 153
6.1 Summary . . . 153
6.2 Future Work . . . 155
6.2.1 Improving the Virtual Projection Room . . . 155
6.2.2 Improving the Generative Model . . . 155
6.2.3 Improving the Camera Models . . . 157
6.2.4 Exploiting Image Uncertainty . . . 158

A Dynamic Stereoscopic Previz **159**
A.1 DSP Presentation 159
A.2 DSP In Action 160

B Super-Resolved Generated Images **163**
B.1 Results 163

C Results from Unstructured Camera Configurations **171**

Bibliography **179**

1 Introduction

1.1 Motivation

The term *five major arts* denoting architecture, sculpture, painting, music and poetry, was introduced by the German philosopher Hegel in his "Lectures on Aesthetics" (Hegel, 1835). In 1911, Ricciotta Canudo in his manifesto "The Birth of the Sixth Art" (Canudo, 1993) claimed that cinema was a new art: *a superb conciliation of the Rhythms of Space (architecture, sculpture, painting) and the Rhythms of Time (music and poetry)* [1]. For over a hundred years, cinematographers have developed artistic ways to convey the *Rhythms of Space* with a 2 dimensional motion picture, using well-known depth cues, e.g. perspective, depth of field, or relative size of objects. Although Stereoscopic Cinema is as old as "2D cinema", its development has taken considerably more time, mainly due to the physiological constraints of the human ocular system. To create the optical illusion of depth from stereopsis, two slightly different images are shown to each eye. However, this optical illusion may create visual fatigue and/or visual discomfort. Poor acquisition or projection configurations deviating from the ideal ones lead to poor stereoscopic viewing experience. Audience complaints about headaches or sickness after a stereoscopic film projection have been common among the audience for decades. With the arrival of the digital images, most problems arising at the acquisition stage can be solved by post-processing the images. In addition, advances in the acquisition devices, such as motorized rigs precisely controlling the cameras positions, as well as advances in the projection technologies, have made possible to create pleasant stereoscopic viewing experiences in 3D cinemas and televisions. Now that technical progress has made 3D cinema and television a reality, artists should be able to explore new narratives, which take advantage of the optical illusion of depth from stereopsis in the storytelling.

In this thesis we review the causes of visual fatigue and visual discomfort and perform a geometric study of the mathematical constraints associated to each

[1] The cinema became the "Seventh Art" when Canudo added the "dance" as the sixth art, the third Rhythmic art combining music and poetry. (Bordwell, 1997)

phenomenon. These constraints define the limits for an artist to create content which provides a pleasant viewing experience. In particular we focus on the case of filming with two cameras equipped with long focal lengths optics. In 3D cinema or television, the use of long focal length optics either creates a "cardboard effect" or causes ocular divergence. The "cardboard effect" creates a poor stereoscopic experience, whereas the ocular divergence is one of the well known causes responsible for the visual fatigue. Because of this reason, artists are limited in the use of long focal length optics and only use them in very few situations. Indeed, our study shows that in most cases it is impossible to acquire images that create an interesting depth from stereopsis and do not create visual fatigue.

At this point, an artistic question arises: what is the living sculpture the director wants to create with the long focal length optics? To answer the question we propose two different approaches to *define* the desired 3D effect, according to two scenarios where a long focal length optic is often used in 2D. In the first scenario the long focal length optics are used to "get closer" to the scene. For some shots, it may be physically very difficult, or even impossible, to place the camera at a precise location. For example, when filming animals in the wild, the presence of the cameras could modify their behavior, or in sports, it is not allowed to place a camera on the field while the game is at play. In the second scenario long focal length optics are used to add perspective deformations to the space, thus distorting the perceived geometry of the acquired 3D scene. This effect provides an important artistic tool for the directors, as they can convey emotions to the spectator with the distortion of the perceived 3D world. An example of this geometric distortion in 2D is the *Vertigo Effect*, *Hitchcock Zoom* or *dolly zoom*, created by Alfred Hitchcock in 1958 in his feature film *Vertigo*. He compensated the backwards movement of the camera by zooming in the image, to keep constant the size of a target object. Objects in front and behind the target object are strongly distorted. The resulting sequence perfectly conveys the terror of heights felt by the hero. The challenge to generate proper stereoscopic images with long focal lengths is the motivation of this thesis, and the research questions addressed in this manuscript belong to the domain of "3D cinematography" (Ronfard and Taubin, 2007, 2010).

1.2 The Research Problem

To generate suitable stereoscopic images corresponding to these scenarios, images are acquired first and then novel virtual views are rendered. The generic term for these kind of techniques is Image-Based Rendering (IBR). These techniques proceed mainly in two stages. In the first stage, the input images are warped into the target view, i.e. the information acquired by the input images is transferred into the target view. This transfer is usually done with the help of a geometric approximation of the observed scene. This approximation is referred to as geometric proxy and it can be implicit or explicit. The next stage is the fusion of the warped images. Should a view be preferred over the others? Which criteria could help us to perform this selection without human intervention? In this thesis we address these questions and provide an answer.

IBR has been an active field of research for the last two decades. Some heuristics have been explored and proven to achieve very good results. Yet the combination of these effective heuristics is not straightforward and relies on some parameters. These parameters present two main drawbacks. The first inconvenience is that they are often adjusted depending on the content of the scene, thus requiring a human intervention. The second drawback is that the parameter magnitudes do not represent physical units, as they weight penalties or energy terms. Hence, it is difficult to choose and justify their values.

In this thesis a new Bayesian approach to the problem of novel view synthesis is proposed. A new generative model is contributed, which takes into account the uncertainty of the image warps in the image formation model. As the warps are given by an explicit or implicit geometric proxy, they have physical units. The Bayesian formalism allows us to deduce the energy of the generative model and to compute the desired images as the Maximum a Posteriori estimate. Moreover, the energy equations provide a formalization of the heuristics widely used in IBR techniques. The benefits of this formalization are multiple. First, the formalization provides insights on which physical phenomena could lie behind each heuristics, thus allowing to state the novel view synthesis problem in an intrinsically parameter-free form: the parameters of the proposed method have physical units and can be measured from the input images. Furthermore, the use of the geometric uncertainty, allows the method to adapt to different qualities of geometric proxy, automatically leveraging the contributions of each camera. Areas where the geometric proxy is more reliable are automatically treated differently from areas where the geometric proxy is less reliable without human intervention. Besides, the proposed generative model addresses the problem of super-resolution, allowing to render images at a higher resolution than the initial ones. The method outperforms state of the art image-based rendering techniques on challenging datasets.

The research questions addressed in this thesis can be summarized as follows. The first question is: *how can we generate stereoscopic shots with long focal length?* The answer to this question leads to the next research question: *given N warped views of a scene, how can we automatically blend the multiple shots into one?* The answer to the second question allows us to formulate and answer our last research question: *how should we place the cameras to generate the stereoscopic shots with long focal length?*

1.3 Contributions

The main contributions of this thesis go beyond the state of the art of stereoscopic cinematography and IBR and they are the following:

The Virtual Projection Room. A new visualization tool, allowing to better understand the 3D distortions of a 3D scene when acquired with a stereoscopic pair of cameras and projected in a projection room in front of a spectator. The proposed approach presents a 3D synthetic view of the spectator in the

projection room. The user can *see* the perceived depth from stereopsis. So, the virtual projection room provides an interactive manipulation of the acquisition and projection parameters, thus allowing a fast exploration of the different acquisition and projection configurations. This contribution is shown in Sec. 3.3. Moreover, the *virtual projection room* was integrated into the software *Dynamic Stereoscopic Previz* that is presented in Appendix A. This shooting simulator was used in the actual shooting of a stereoscopic short film: "Endless Night".

A new Bayesian approach formalizing the principles of Image Based Rendering. The key theoretical contribution of the proposed method is the systematic modeling of the error introduced in the Lambertian image formation process via the inaccuracy in the estimates of the geometric proxy. We call this inaccuracy *depth uncertainty*, referring to the depth estimates from the input images. In addition to this error, we also consider the image sensor noise, commonly modeled as Gaussian. We extensively analyze the theoretical implications of the obtained energy, discussing the formal deduction of the state of the art heuristics from our model. This work provides the first Bayesian formulation explicitly deriving the heuristics of Buehler *et al.* (2001). The equations obtained using the Bayesian formalism have the advantage of being essentially parameter-free.

From a practical point of view, we numerically evaluate the performance of our method for two cases. First we address a simplified camera configuration where all viewpoints are in a common plane, which is parallel to all image planes. This configuration is known as the Lumigraph (Gortler *et al.*, 1996). For this configuration we compare our results to the best existing method within the Bayesian framework (Wanner and Goldluecke, 2012). In a second set of experiments we deal with the generic, unstructured configuration as proposed in Buehler *et al.* (2001). For this configuration we implemented the generic extension of Wanner and Goldluecke (2012) as well as the method proposed by Buehler *et al.* (2001). We compare our results to both methods.

Experimental results show that we achieve state of the art results with regard to objective measures on public datasets. Moreover, we are also capable of addressing super-resolution, capitalizing on the general framework established in Wanner and Goldluecke (2012). The new model is not without a price, since its optimization is less straightforward. However, existing methods allow us to overcome this difficulty.

The Stereoscopic Zoom. The last contribution of this thesis is to analyze the 3D distortions that arise when acquiring stereoscopic images with long focal lengths and to propose two approaches to overcome these distortions. The proposed approaches are an answer to the actual intentions of the directors in the use of the long focal lengths in the 2D cinema or television: to *get closer* to the scene or to *add perspective deformations* of the acquired scene. For the scenario where the director wants to *get closer* to the scene we propose to generate virtual novel views at the desired camera locations (Sec. 5.1). For the scenario where the director wants to *introduce perspective deformations* we propose to distort the acquired 3D world, in order to generate the desired stereoscopic images (Sec. 5.2).

Both methods benefit from the Bayesian approach to the IBR problem. We present a scenario where both approaches can be considered and discuss its advantages and limitations.

1.4 Thesis Outline

Chapter 2. In Chapter 2 the factors leading a human to perceive depth are reviewed. The monoscopic and stereoscopic depth cues, as well as the physiological constraints leading to visual discomfort and visual fatigue are presented.

Chapter 3. In Chapter 3 a geometric approach to the depth perception from stereopsis is presented. The concepts described in Chapter 2 are mathematically formalized and a new visualization tool, the "virtual projection room", is presented. This software allows to better understand the complex transformation between the acquired 3D scene and the 3D scene perceived by the spectator in the projection room.

Moreover, the geometric distortions arising when changing the projection configuration are illustrated, and the state of the art approaches that address the problem are reviewed. The geometric distortions arising when using acquisition cameras with long focal lengths are also shown, and it is explained why the limitations of the existing methods prevent to obtain the desired results. Finally, two IBR approaches to create stereoscopic images with long focal lengths are derived in this chapter.

Chapter 4. In Chapter 4 the existing IBR methods are reviewed. Then our novel generative model for the image formation process is presented and its associated energy deduced. The performance of this new model is demonstrated by means of two sets of experiments. First, the simplified camera setup corresponding to the Lumigraph (Gortler *et al.*, 1996) is considered. In this camera setup the obtained equations are simpler. Then the general case corresponding to the Unstructured Lumigraph (Buehler *et al.*, 2001) is analyzed. The creation of the necessary input is detailed and the performance of the proposed method illustrated. The benefits and limitations of our own approach are discussed, and to conclude the chapter, the relation of the proposed approach with the *desirable properties* of Buehler *et al.* (2001) is analyzed.

Chapter 5. In Chapter 5 two approaches to generate stereoscopic images with long focal lengths are presented. The first is based on the director's intention to *get closer* to the scene while the second is based on the director's intention to create *perspective distortions* of the scene. A scenario where both approaches can be used is presented and the actual camera positions are deduced according to the IBR approach presented in Chapter 4. The Chapter is concluded with a discussion of the advantages and limitations of both approaches.

Chapter 6. Chapter 6 closes this thesis by summarizing the main contributions. The impact of our work and some of the learned lessons are discussed. Finally, leads on how future work can address the remaining issues are proposed.

2 Depth Perception and Visual Fatigue

In this chapter we briefly present the process of depth perception when viewing stereoscopic moving pictures, which has been extensively covered in the literature (Gibson, 1950; Lipton, 1982; Todd, 2004; Devernay and Beardsley, 2010). We review the *depth cues* leading to the perception of depth, the consequences of contradictory or inconsistent depth cues, and the causes leading to visual fatigue when viewing stereoscopic motion images. The goal of this chapter is to illustrate how the perceptual human factors make the acquisition and projection of stereoscopic images a much more constrained problem than the acquisition and projection of traditional 2D moving pictures.

2.1 Depth Cues

In this section we review the visual features producing depth perception which are known as *depth cues*. They can be grouped in two classes, the monoscopic depth cues, present in a 2D representation of the world, and the stereoscopic depth cues, arising when using a binocular system.

2.1.1 Monoscopic Cues

Lipton (1982) proposes seven monoscopic depth cues which are well know to *encode* the depth in a 2D representation. We illustrate them in Fig. 2.1.

Retinal Image Size Larger retinal images tell us that the object is closer, because objects closer to the eye are seen as larger.

Perspective or Linear Perspective Objects diminish their size as they recede from the observer. For example, parallel railroads seem to converge at the horizon.

Interposition or Overlapping One object in front of another prevents us from seeing the one behind. A teacher in front of the blackboard cuts part of the view of the blackboard and must, therefore, be closer to the student than the blackboard.

Aerial perspective Atmospheric haze provides the depth cue of aerial perspective. In a very hazy day, the mountain is barely visible in the glare of the haze illuminated by the setting sun. The haze intervening between the observer and the mountain makes the mountain look far away.

Light and Shade Cast shadows provide an effective depth cue, as does light coming from one or more directions modeling an object.

Textural Gradient This cue is discussed at great length by Gibson (1950). The leaves of a tree are clearly discernible up close, but from a distance the texture of the leaves becomes less detailed.

Motion Parallax When the point of view changes, objects near the viewer have a larger image displacement than objects being far away.

Depth of Field Although it is usually forgotten in the list of monoscopic depth cues (Lipton, 1982), the *depth of field* or *retinal image blur* is a monoscopic depth cue (Held *et al.*, 2010). Objects with different blur size are perceived at different depths.

Lipton (1982) also claims that the *accommodation*, the muscular effort involved in focusing, could provide a feedback or proprioceptive mechanism for gauging depth. However, it is not clear from psychophysics experiments whether this should be considered as a depth cue or not (Devernay and Beardsley, 2010).

2.1.2 Stereoscopic Depth Cues

The fact that we are looking at a scene using our two eyes brings two additional physiological depth cues (Lipton, 1982): convergence and disparity.

Convergence The lens of each eye projects a separate image of objects on each retina. In order for these to be seen as a single image by the brain, the central portion of each retina must "see" the same object point. The muscles responsible for this convergence, the inward or outward rotation of the eyes, may provide distance information.

Disparity When eyes converge on an object in space, it is seen as a single image, and all other objects, in front or behind the point of convergence can be seen to be double images. The disparity is the difference between the two retinal image positions of a scene point.

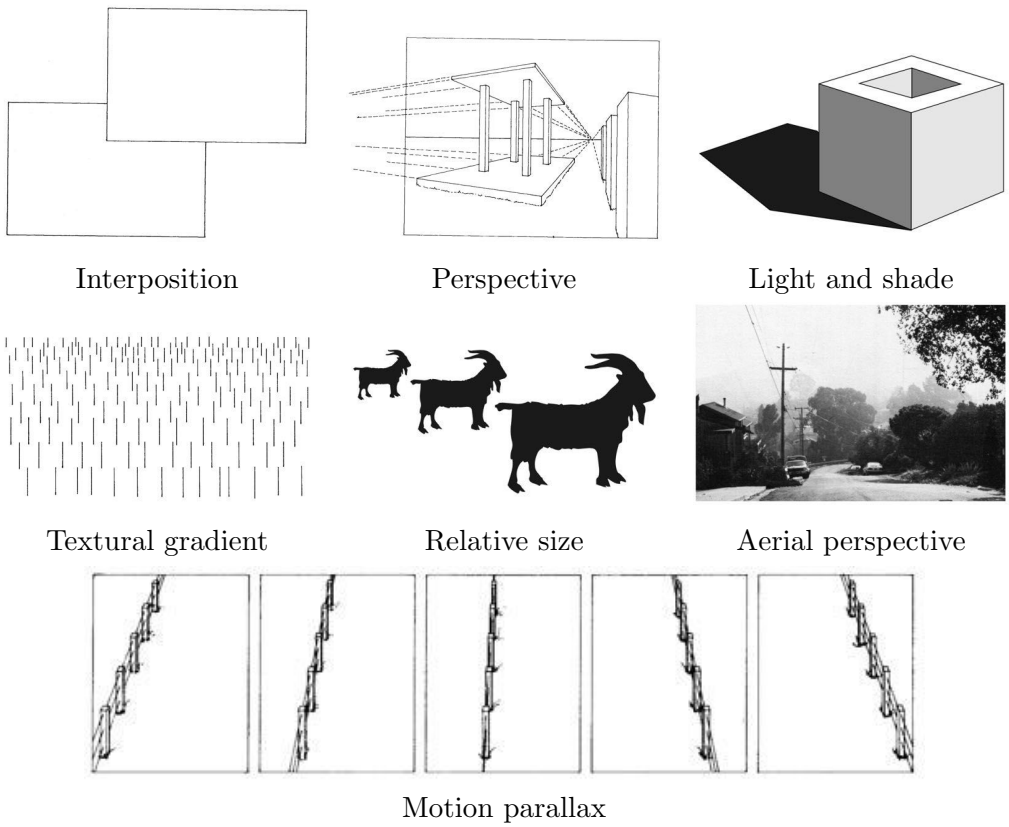


Fig. 2.1: *Illustrations of the seven monoscopic depth cues described by Lipton (1982): interposition, perspective, light and shade, textural gradient, relative size, aerial perspective and motion parallax. Images reproduced from Lipton (1982) and Devernay and Beardsley (2010).*

These *steresocopic depth cues* are used by the perception process called stereopsis, giving a sensation of depth from two different viewpoints. The term "stereopsis" was first described by Sir Charles Wheatstone in Wheatstone (1838). In Fig. 2.2 we illustrate the perceived depth from stereopsis.

All those depth cues, one by one, and in their combination, allow us to perceive depth. Special care should be taken when creating new steresocopic views. The depth "described" individually by each depth cue should be coherent with the depth described by the others, as formulated by Lenny Lipton: *"Good 3D is not just about setting a good background. You need to pay good attention to the seven monocular cues (...) Artists have used the first five of those cues for centuries. The final stage is depth balancing."* Conflicting or inconsistent depth cues can lead to a poor viewing experience.

2.1.3 Conflicting Depth Cues

Conflicting depth cues arise when two different cues provide depth information pointing in different directions. In 1754 William Hogarth provided a very nice illustration (see Fig. 2.3) showing the importance of coherent depth cues by

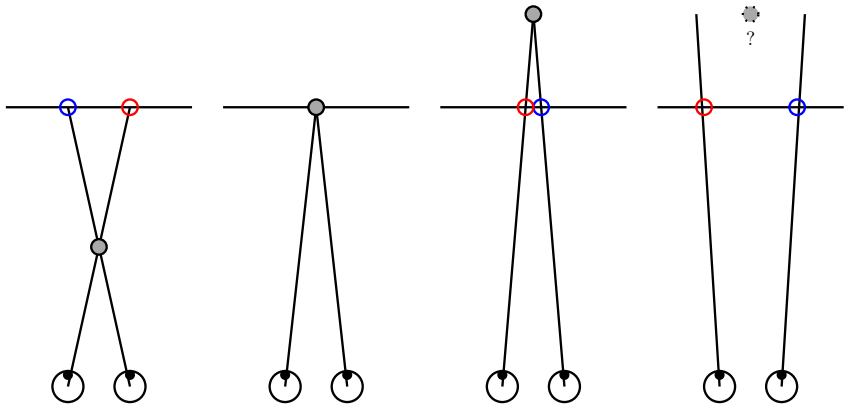


Fig. 2.2: *Perceived depth from stereopsis. From left to right, the 3D point is perceived, in font of the screen (positive disparity), at the depth of the screen (zero disparity) and behind the screen (negative disparity). In the last configuration, ocular divergence arises: the optical rays intersect behind the spectator.*

contradiction. Let us focus on two examples of conflicting depth cues, involving the treeline in the center of the image behind the bridge.

The first conflict we are interested in, is the interposition of the flag with the trees. Because of the relative size of objects, the trees seem to be far compared to the flag. But the trees interpose the flag, thus they should be in front of the flag.

The second conflict is a contradiction between relative size and perspective. Because of the perspective in the tree line, the left trees seem to be farther away than the right trees. However, as the left trees are bigger in size, they seem to be enormous compared to the right ones.

Similar issues arise when the stereopsis depth cue is in contradiction with other depth cues.

The *window violation* is a well known issue (Mendiburu, 2009) arising when the depth from stereopsis and interposition are in contradiction. When an object in front of the screen is cut by the border of the image, the stereopsis depth cue tells us that the object is in front of the screen border. However, the border "cuts" the object, thus it must be in front of it. A way to solve this issue is the use of *floating windows* (Mendiburu, 2009). By adding a black border to an image, the perceived depth from stereopsis of the image border can be "pushed" forward. Thus the interposition is coherent with the depth from stereopsis.

Reverse stereo is another well know issue where the sterescopic depth cues are in contradiction with the monoscopic depth cues. The pseudoscope invented by Sir Charles Wheatstone is a device which switches the viewpoints of both images. The sterescopic cues are then reversed, while the monoscopic cues are preserved. The viewer experiences a 3D perception of the scene, but depth cues are in conflict. For example, similarly to the tree line in Fig. 2.3, the relative size of objects indicates a depth cue which is in contradiction with the (reversed) perceived depth from stereopsis.

Fig. 2.3: *"Whoever makes a DESIGN without the Knowledge of PERSPECTIVE will be liable to such absurdities as are shown in this Frontispiece." William Hogarth 1754.*

2.1.4 Inconsistent Depth Cues

While two conflicting depth cues indicate contradictory depths, inconsistent depth cues indicate different amounts of depth in the same direction (Devernay and Beardsley, 2010). While they are in general less disturbing than conflicting depth cues, they can lead to a poor stereoscopic experience and they may even spoil the sensation of reality (Yamanoue *et al.*, 2006). Two well known effects creating inconsistent depth cues are the *cardboard effect* and the *puppet-theater effect*.

The *cardboard effect* arises when some depth from stereopsis is clearly perceived between the elements on the observed scene, but the elements themselves lack depth. They appear as flat, or as a drawn on a cutout cardboard. This effect is common in anaglyph comic books of the fifties, because each element was drawn in 2D, and then horizontally offset to give an illusion of 3D. Although elements are perceived at different depths, they are still flat 2D drawings. In this case the inconsistency arises from the monoscopic depth cues (light and shade, relative size, perspective, . . .) and the depth from stereopsis: the viewer perceives some depth from stereopsis between the elements; the monoscopic cues point in the same direction but the depth from

stereopsis of the elements themselves is reduced.

The *puppet theater effect* or *pinching effect* is another disturbing effect, where elements of the scene look unnaturally small (Yamanoue *et al.*, 2006). This effect is driven by an inconsistency between the monoscopic depth cue *relative size of objects* and the perceived depth from stereopsis. The depth estimated from the the relative size of an object in the foreground and an object in the background is not consistent with the perceived depth from stereopsis. This effect appears when the elements of the scene suffer a size distortion which is different depending on the depth of the object. Note that in general, it is only possible to perceive the depth from the monoscopic cue *relative size of objects* for known objects. As stated by Yamanoue *et al.* (2006), *no one can evaluate the size of an object that has never been seen before*. However, once the viewer gets familiar with the size of the object, the effect can (and will) arise.

As noted by Devernay and Beardsley (2010), both effects (*cardboard effect* and *puppet-theater effect*) can be easily avoided if one is in total control of the shooting configuration, including the camera placement. However, if the shooting configuration is constrained, the effects may appear.

2.2 Visual Comfort and Visual Fatigue

Visual fatigue has been for certain the main cause of the failure of stereoscopic cinema in the past century. Visual fatigue, also named eyestrain, can manifest in a wide range of visual symptoms, e.g. tiredness, headaches, dried mucus, or tears around the eyelids among others (Ukai and Howarth, 2008). Visual comfort is used interchangeably with visual fatigue in the literature, but, as stated by Lambooij *et al.* (2007), a distinction should be made. Visual fatigue can be measured as a decrease of performance of the human visual system, whereas visual comfort is subjectively self-reported.

In this section we review the sources of visual fatigue when viewing stereoscopic motion images, which are today well known. They can be listed as *stereoscopic image asymmetries* (Kooi and Toet, 2004) the *vertical disparities* (Allison, 2007; Lambooij *et al.*, 2007), the *crosstalk* (Yeh and Silverstein, 1990; Kooi and Toet, 2004), the *horizontal disparity limits* (Yeh and Silverstein, 1990), and the *vergence-accomodation conflict* (Hoffman *et al.*, 2008; Shibata *et al.*, 2011; Banks *et al.*, 2013). They can be arranged in two groups. In the first one we have the *stereoscopic image asymmetries*, the *vertical disparities* and the *crosstalk* which constrain the mechanical systems (acquisition and projection) but do not constrain the stereoscopic artistic choices, i.e. the depth of a scene element. In the other group we have the *vergence-accommodation conflict* and the *horizontal disparity limits* which constrain the depth at which a scene element can be projected to, or the relative depth between scene elements. In this work we refer to those stereoscopic artistic choices as the *stereoscopic mise-en-scene*.

The results obtained by Kooi and Toet (2004) show that almost all stereoscopic image *assymetries* seriously reduce the visual comfort. Those asymmetries arise

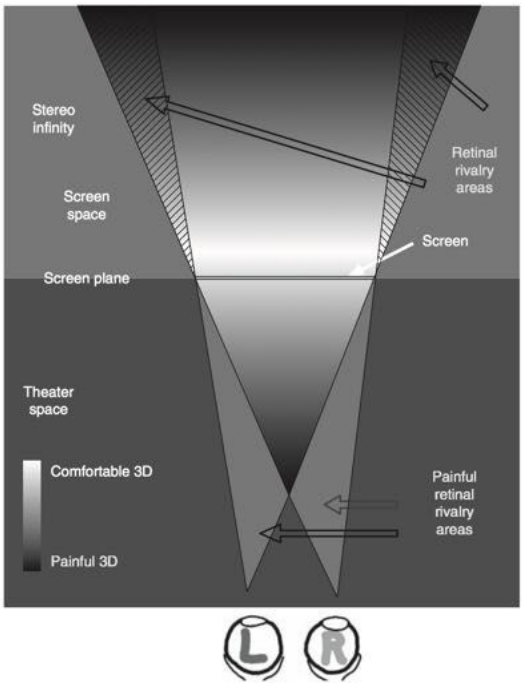


Fig. 2.4: *The stereoscopic comfort zone. Reproduced from Mendiburu (2009).*

from imperfections, either in the acquisition setup (camera alignment, optics mismatch, camera desynchronization, ...) or in the projection setup (projector alignment, optics mismatch, projector desynchronization, ...). They are of course very important and need to be accounted for. However, they rely on purely mechanical or software technical solutions (e.g. camera and projector alignment) which are addressed in the literature (Zilly *et al.*, 2011, 2010). They do not constrain the depth of the scene elements. Similarly, *vertical disparities* in the acquired images can be eliminated with image rectification (see Sec. 3.1.5), and do not constrain the *stereoscopic mise-en-scene* either.

The *crosstalk* (or *crossover* or *ghosting*) arises from the inability of the projection system to properly filter the left and right images. Light of the left image leaks to the image seen by the right eye, and vice-versa, thus creating artifacts known as *ghosts*. The *crosstalk* has an artistic impact on how elements at different depth should be lighted (Mendiburu, 2009), but does not affect the range of depths where the element can be displayed at.

In our work we are interested with the sources constraining the *stereoscopic mise-en-scene*: the *vergence-accomodation conflict* and *horizontal disparity limits*. In Fig. 2.4 we illustrate a scheme of the comfortable depth perception zones, usually called "comfort zones".

2.2.1 Vergence-Accomodation Conflict

When looking at an object in the real world, our eyes toe in to converge at the distance of the observed object. This distance is known as the *vergence distance*. At the same time, our eyes accommodate to bring the image of the object at that depth to sharp focus. This distance is known as the *focus distance*. As both distances are equal in natural viewing, convergence and accommodation are neurally coupled (Fincham and Walton, 1957). This coupling allows an increased response speed: accommodation and vergence are faster with binocular vision than with monocular vision (Cumming and Judge, 1986).

However, when viewing stereoscopic motion images, the viewer accommodates at the screen distance, while its ocular system convergence is done at the distance where the scene object is presented. Because of the strong coupling in the visual system, this difference creates a conflict, which is known as the *vergence-accommodation conflict*. The resolution of the conflict by the human visual system may create visual fatigue (Hoffman *et al.*, 2008; Lambooij *et al.*, 2009; Shibata *et al.*, 2011; Banks *et al.*, 2013).

This phenomenon has been studied in optometry and ophthalmology. The goal is to establish the zone of clear single binocular vision (ZCSBV), which is the set of vergence and focal stimuli that the patient can clearly see while maintaining the binocular fusion. Shibata *et al.* (2011) and Banks *et al.* (2013) provide a very complete overview of the historical evolution of the estimation of the ZCSBV, from the first measures from Donders in 1864 and the *Percival's zone of comfort* established in 1892, to the nowadays measured boundaries. To our knowledge, they contribute the most recent experimental results establishing the boundaries of the *vergence-accomodation conflict*, that we reproduce in Fig. 2.5.

Two important points arise from the *vergence-accommodation conflict*. The first is that the amount of 3D space available is limited by the comfort zone. Placing scene elements out of the comfort zone will most probably create visual fatigue and the viewer may experience diplopia, which is the inability to fuse stereoscopic images. The second remark is that the comfort zone depends on the viewing distance of the viewer. Moreover, as the viewing distance is often related to the size of the screen (see Sec. 3.2.7), we can extrapolate that the depth limits of the comfort zone are different depending on the size of the screen. Not only the 3D space available is limited, but the limits change with the size of the screen.

2.2.2 Horizontal Disparity Limits

Although the *horizontal disparity limits* are related to the *vergence-accomodation conflict*, they do not represent the same thing. We saw that the studies addressing the *vergence-accomodation conflict* focused on the estimation of the ZCSBV. However, the human visual system is not capable to fuse at the same time objects at very different depths. Even if a foreground and a background objects are inside the ZCSBV, fusing both of them at the same time may be difficult.

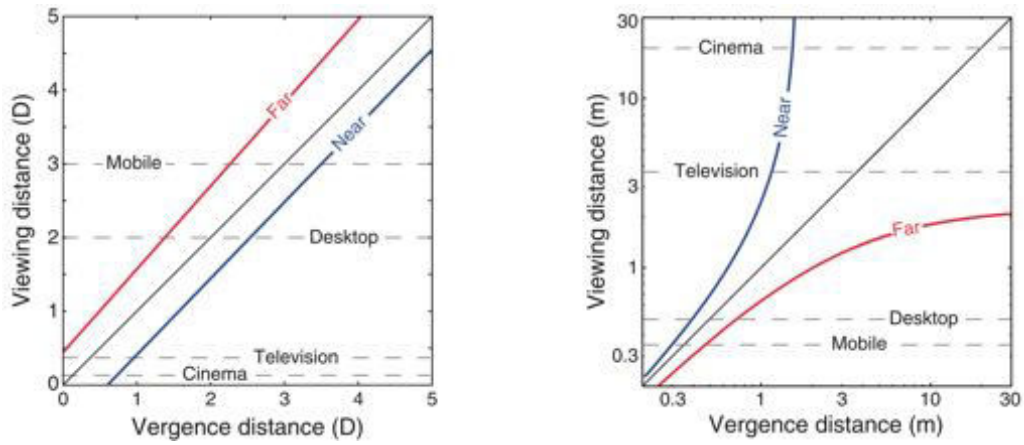


Fig. 2.5: *Figure reproduced from Shibata et al. (2011). The graphics represent the empirically estimated vergence-accomodation conflict limits. In the left graphic, distance is represented in Diopter units (D), which are the inverse of meters: $D = \frac{1}{m}$. In the right graphic the same graphic is presented in metric units. The boundaries of the vergence-accommodation conflict depend on the viewing distance. The dashed horizontal lines represent typical viewing distances for mobile devices, desktop displays, television, and cinema. The comfort zone gets smaller as the viewing distance decreases.*

Mendiburu (2009) introduces three practical concepts: the *stereo real state* denoting the amount of 3D space available in the projection room, the *depth bracket* denoting the portion of 3D used in a shot or sequence, and the *depth position* denoting the placement of the depth bracket inside the stereo real state. Yano *et al.* (2004) showed that stereoscopic images with a bigger *depth bracket* than the human depth of field cause visual fatigue. This finding is coherent with the strong relation between the horizontal disparity limits and the human depth of field boundaries found by Lambooij *et al.* (2009). Although exact values might slightly differ depending on the work, it is commonly accepted that the depth of field guides the horizontal disparity limits.

Although using an excessive depth bracket may create visual fatigue, some artistic effects may ask an excessive disparity range. A common practice in the stereoscopic film industry is to create a *depth script* or *depth chart*, a time line with the *depth bracket* of the shots and sequences (see Fig. 2.6). In order to compensate for an excessive *depth bracket*, a low 3D sequence, or "rest area", can allow the audience's visual system to recover from the effort (Mendiburu, 2009; Liu *et al.*, 2011).

The last, but not less important disparity limit, is the ocular divergence (Spottiswoode *et al.*, 1952). If a viewer tries to fuse a disparity on the screen bigger than the human interaxial, their eyes will diverge (see Fig. 2.2). Images creating a low divergence angle (up to 1°) can still be fused (Shibata *et al.*, 2011), although they may create visual fatigue if divergence occurs for a long time. Images creating a divergence angle higher than 1° are likely to create diplopia (Shibata *et al.*, 2011).

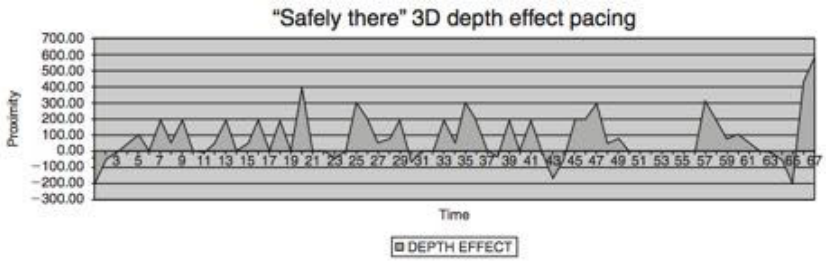


Fig. 2.6: *An example of depth chart. Figure reproduced from Mendiburu (2009).*

2.3 Summary

In this chapter we have seen that shooting a stereoscopic movie involves more constraints than in the traditional 2D cinema. Not only the monoscopic depth cues must be coherent, but the supplementary stereoscopic depth cues must point in the same direction. We have introduced the *window violation*, the *puppet-theater effect*, and the *cardboard effect* as well as the *vergence-accomodation conflict* and the *horizontal disparity limits*. In the next chapter we formalize these concept into mathematical constraints.

3

Stereoscopic Filming: a Geometric Study

In this chapter we present a geometric approach to the depth perception from stereopsis. We first introduce the mathematical models and notations used in the rest of the chapter. Then we mathematically formalize the concepts described in the previous chapter and derive the constraints on the acquisition setup to avoid the visual discomfort and visual fatigue. We present a new visualization tool, the "virtual projection room", allowing to better understand the complex transformation between the acquired 3D scene and the 3D scene perceived by the spectator in the projection room. We illustrate the geometric distortions arising when changing the projection configuration, and review the state of the art approaches that address the problem. Those methods introduce the concept of *disparity mapping*, a clever function allowing to reduce those distortions. We analyze the impact of the *disparity mapping function* into the mathematical formalization of the constraints. We also illustrate the geometric distortions arising when using acquisition cameras with long focal lengths, and explain why the limitations of the existing methods prevent to obtain the desired results. We derive two image-based rendering approaches to create stereoscopic images with long focal lengths.

3.1 3D Transformations and Camera Matrices

In this section we introduce the mathematical models and notations associated to projective geometry applied to computer vision. We introduce the pinhole camera model, its associated 3D to 2D camera projection matrix, and the reconstruction matrix, a 3D to 3D transformation. We then consider the two camera case and introduce the epipolar geometry. We detail the configuration of two rectified cameras, which is key to stereoscopic filming and projection.

This section assumes some familiarity of the reader with projective geometry

applied to computer vision. For a much more detailed introduction we refer the reader to the reference books Faugeras (1993), Forsyth and Ponce (2002), Hartley and Zisserman (2004) and Szeliski (2010). We choose the latest (Szeliski, 2010) as reference book for our notations.

3.1.1 3D Translations and Rotations

A 3D translation in space is given by a 3 component vector $\boldsymbol{t}$, and we write it as $\boldsymbol{x}' = \boldsymbol{x} + \boldsymbol{t}$, or as a 3×4 matrix product in the form

$$\boldsymbol{x}' = [\boldsymbol{I}|\boldsymbol{t}]\bar{\boldsymbol{x}}, \tag{3.1}$$

where $\boldsymbol{I}$ is the 3×3 identity matrix, and $\bar{\boldsymbol{x}} = (x, y, z, 1)$ is the augmented vector of $\boldsymbol{x} = (x, y, z)$.

A 3D rotation in space is described using a 3×3 matrix $\boldsymbol{R}$, an orthogonal matrix ($\boldsymbol{R}^\top = \boldsymbol{R}^{-1}$) with $\det \boldsymbol{R} = 1$. This matrix can be parametrized using either *Euler angles*, the *exponential twist* or *unit quaternions*. The *Euler angles* are three angles $(\theta_x, \theta_y, \theta_z)$, each one describing a 3 rotation around the x-, y- and z-coordinate axis. The *exponential twist* is parametrized by a rotation axis $\hat{\boldsymbol{n}}$ and an angle θ, and the *unit quaternions* are often written as $\boldsymbol{q} = (q_x, q_y, q_z, q_w)$. The use of *Euler angles* is in general a bad idea (Faugeras, 1993; Diebel, 2006) because it depends on the order in which the transforms are applied. The choice between the *exponential twist* and the *unit quaternions* is often driven by the application.

A 3D rotation and translation is also known as a *3D rigid body motion* or *3D Euclidean transformation*. We write it as $\boldsymbol{x}' = \boldsymbol{R}\boldsymbol{x} + \boldsymbol{t}$, or

$$\boldsymbol{x}' = [\boldsymbol{R}|\boldsymbol{t}]\bar{\boldsymbol{x}}. \tag{3.2}$$

3.1.2 Perspective 3D to 2D Projection

There exist several types of 3D to 2D projections: orthographic, scaled orthography, para-perspective, perspective and object-centered (Szeliski, 2010). In our work we use perspective, since this more accurately models the behavior of real cameras. In Fig. 3.1 we illustrate a perspective projection of a 3D point $\boldsymbol{x} = (x, y, z)$ into an

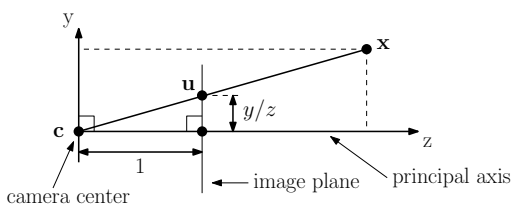


Fig. 3.1: *Scheme of the perspective projection. Illustration adapted from Hartley and Zisserman (2004). A 3D point $\boldsymbol{x} = (x, y, z)$ is projected onto the image plane point $\boldsymbol{u} = (\frac{x}{z}, \frac{y}{z}, \frac{z}{z})$. The distance between the image plane and the camera center $\boldsymbol{c}$ is considered to be 1.*

image plane. The 3D point is projected onto the image plane by dividing it by its z component. We obtain the 3D point $\boldsymbol{x}' = (\frac{x}{z}, \frac{y}{z}, \frac{z}{z})$. We note the homogeneous coordinates of the projected point with a tilde over the vector, e.g. $\tilde{\boldsymbol{x}}' = (\tilde{x}, \tilde{y}, \tilde{w}) = \tilde{w}\ (\frac{x}{z}, \frac{y}{z}, 1)$. As the third coordinate of the 3D point $\boldsymbol{x}'$ is always 1, $\boldsymbol{x}'$ can be considered as the extension into homogeneous coordinates of a 2D point $\boldsymbol{u}$: $\bar{\boldsymbol{u}} = \boldsymbol{x}'$. In homogeneous coordinates the perspective projection of the 3D point $\boldsymbol{x} = (x, y, z)$ has a linear form

$$\tilde{\boldsymbol{u}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \bar{\boldsymbol{x}}. \tag{3.3}$$

We drop the last component of $\boldsymbol{x}$, thus, once the 3D point is projected, it is not possible to recover its distance to the camera.

3.1.3 Pinhole Camera Model

In this manuscript we use the *pinhole camera model* and represent it with its camera matrix $\boldsymbol{P}$. An extensive description can be found in Chapter 6 of Hartley and Zisserman (2004). The basic idea is that a camera establishes a mapping between the 3D points in the world $\boldsymbol{x} \in \mathbb{R}^3$ and the 2D image points $\boldsymbol{u} \in \mathbb{R}^2$ in pixel units.

The camera projection matrix (Hartley and Zisserman, 2004) is a 3×4 matrix $\boldsymbol{P}$, such that

$$\tilde{\boldsymbol{u}} = \boldsymbol{P}\bar{\boldsymbol{x}}. \tag{3.4}$$

The camera projection matrix $\boldsymbol{P}$ can be decomposed into the matrices $\boldsymbol{K}$ and $(\boldsymbol{R}|\boldsymbol{t})$:

$$\boldsymbol{P} = \boldsymbol{K}(\boldsymbol{R}|\boldsymbol{t}). \tag{3.5}$$

The 3×3 matrix $\boldsymbol{K}$ is called the *intrinsic camera parameters* and the matrix $\boldsymbol{R}$ and the vector $\boldsymbol{t}$ define the *extrinsic camera parameters*. The 3×3 matrix $\boldsymbol{R}$ is the rotation of the camera with respect to the 3D world. The 3 dimensional vector $\boldsymbol{c}$ is the position of the optical center in the 3D world. The 3 dimensional vector $\boldsymbol{t} = -R\boldsymbol{c}$ is the position of the origin of the world in the camera frame. The transformation $(\boldsymbol{R}|\boldsymbol{t})$ transforms a 3D point in the world frame into a 3D point in the

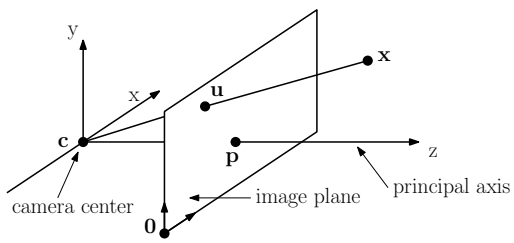


Fig. 3.2: *Scheme of the pinhole camera model projection. Illustration adapted from Hartley and Zisserman (2004). A 3D point $\boldsymbol{x}$ is projected onto the pixel coordinates $\boldsymbol{u}$. $\boldsymbol{0}$ is the origin of the pixel coordinates and $\boldsymbol{p}$ is the principal point of the camera in pixel coordinates.*

camera frame. The homogeneous coordinates normalization does the perspective projection into the projection image plane. The 3×3 matrix $\boldsymbol{K}$ then transforms points on the projection image plane, into the pixel domain. A convention (Szeliski, 2010) is to write the intrinsic parameters $\boldsymbol{K}$ in an upper-triangular form:

$$\boldsymbol{K} = \begin{pmatrix} f_x & s & p_x \\ 0 & f_y & p_y \\ 0 & 0 & 1 \end{pmatrix}. \tag{3.6}$$

The entry s encodes any possible skew between the sensor axes due to the sensor not being mounted perpendicular to the principal axis. In our work we usually set $s = 0$. Although pixels are normally rectangular instead of square (Forsyth and Ponce, 2002), for the sake of simplicity in this work we assume them to be square. The pixels coordinates have then the same scale factor f: $f_x = f$ and $f_y = f$. The 2 dimensional vector $\boldsymbol{p} = (p_x, p_y)$ denotes the optical center expressed in pixel coordinates. It is usually set to the half of the width and height of the image, but in our work it will be useful to consider a *decentered* optical center, as we will see in Sec. 3.1.5.

Hence in this work we use the *intrinsic camera parameters* in the form

$$\boldsymbol{K} = \begin{pmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{pmatrix}. \tag{3.7}$$

3.1.4 Epipolar Geometry Between Two Cameras

Two cameras is the minimal vision system allowing to infer the depth of the observed scene from the images. If we are capable to associate two points in the images, we can deduce the 3D location of the imaged point by triangulation. The geometry defined by two cameras is known as *epipolar geometry*. Let us introduce the basic definitions.

Let us consider two cameras and the 3D point $\boldsymbol{q}$ as illustrated in Fig. 3.3. The projection of the optical center $\boldsymbol{c}_0$ into the camera 1, is known as the *epipole* $\boldsymbol{e}_1$. The projection of the optical center $\boldsymbol{c}_1$ into the camera 0, is known as the *epipole* $\boldsymbol{e}_0$. The pixel $\boldsymbol{x}$ on the camera $\boldsymbol{c}_0$, projects to an *epipolar line segment* in the other image. The plane defined by the optical centers $\boldsymbol{c}_0$, $\boldsymbol{c}_1$ and the pixel $\boldsymbol{x}$ (or the 3D point $\boldsymbol{q}$) is known as the *epipolar plane*. For a given pixel $\boldsymbol{x}$, the epipolar lines $\boldsymbol{l}_i$ define the range of possible locations the pixel may appear at in the other image. An interesting configuration is when the epipolar lines are horizontal in the image. This configuration is called *rectified configuration*. For example, an advantage of this configuration is that it allows the search algorithms to perform a one dimensional search instead of a bi-dimensional search in the image. The pre-warp to transform two generic cameras into a rectified configuration is know as the *camera rectification* and its computation is well known in the literature (Loop

and Zhang, 1999; Fusiello *et al.*, 2000; Faugeras and Luong, 2004; Hartley and Zisserman, 2004; Szeliski, 2010). Fig. 3.4 illustrates the obtained results with the method proposed by Loop and Zhang (1999). A two camera configuration where both cameras are looking in a similar direction is also known as *stereoscopic camera* or simply *stereo*. In these configurations, cameras are usually addressed as the *left* and the *right* cameras.

3.1.5 Two Rectified Cameras

Let us now construct two projection camera matrices $\boldsymbol{P}_l$ and $\boldsymbol{P}_r$, with their intrinsic and extrinsic parameters $(\boldsymbol{K}_l, \boldsymbol{R}_l, \boldsymbol{t}_l)$ and $(\boldsymbol{K}_r, \boldsymbol{R}_r, \boldsymbol{t}_r)$. The sub-indexes l and r stand for the *left* and the *right* cameras respectively. The parameters of two rectified cameras are related. Two rectified cameras have the same orientation in the world, their rotation matrices are equal: $\boldsymbol{R}_l = \boldsymbol{R}_r$. Without loss of generality we can assume them to be the identity matrix $\boldsymbol{R}_l = \boldsymbol{R}_r = \boldsymbol{I}$ (by counter-rotating the world with $\boldsymbol{R}^{-1}$). We can also assume that the left camera is centered at the origin of the 3D world: $\boldsymbol{c}_l = \boldsymbol{0}$, and thus $\boldsymbol{t}_l = -\boldsymbol{R}\boldsymbol{c} = \boldsymbol{0}$. The segment between the cameras two optical centers is parallel to the image plane, and aligned with the x-coordinate. The distance between the optical centers of the cameras is usually called *baseline*. In the rest of the manuscript we note the *baseline* with the scalar b. Thus we can write $\boldsymbol{c}_r = (b, 0, 0)$, and $\boldsymbol{t}_r = (-b, 0, 0)$. We have all the extrinsic parameters of the rectified cameras. The intrinsic camera parameters are f_l, f_r, $\boldsymbol{p}_l$ and $\boldsymbol{p}_r$. Two rectified cameras have the same image plane, so their focal length is the same $f_l = f_r = f$. Although the choice of the principal points is not constrained by the rectified configuration, a convenient choice is to set the same y-coordinate q for both principal points. The choice of the x-coordinate of the principal point has an impact on the stereo camera system. Our principal points are $\boldsymbol{p}_l = (p_l, q)$ and $\boldsymbol{p}_r = (p_r, q)$.

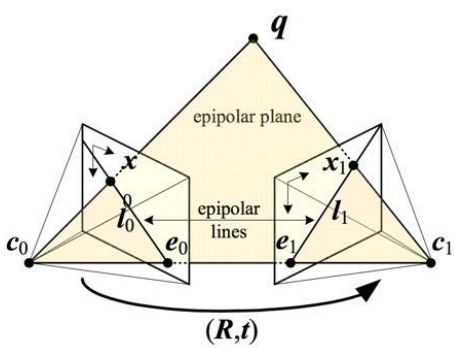


Fig. 3.3: *Figure reproduced from Szeliski (2010) describing the epipolar geometry between two cameras.*

a) b)

Fig. 3.4: *We illustrate an example of image rectification with the algorithm from Loop and Zhang (1999). a) the input pair of images with a set of epipolar lines. b) rectified image pair so that epipolar lines are horizontal and in vertical correspondence. Figure reproduced from Szeliski (2010).*

The obtained left camera parameters are

$$\boldsymbol{K_l} = \begin{bmatrix} f & 0 & p_l \\ 0 & f & q \\ 0 & 0 & 1 \end{bmatrix}, \qquad \boldsymbol{R_l} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad \boldsymbol{t_l} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \tag{3.8}$$

and the camera matrix $\boldsymbol{P}_l = \boldsymbol{K}_l(\boldsymbol{R}_l|\boldsymbol{t}_l)$ is

$$\boldsymbol{P_l} = \begin{bmatrix} f & 0 & p_l & 0 \\ 0 & f & q & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{3.9}$$

The right camera parameters are

$$\boldsymbol{K_r} = \begin{bmatrix} f & 0 & p_r \\ 0 & f & q \\ 0 & 0 & 1 \end{bmatrix}, \qquad \boldsymbol{R_r} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad \boldsymbol{t_r} = \begin{bmatrix} -b \\ 0 \\ 0 \end{bmatrix}, \tag{3.10}$$

and the camera matrix $\boldsymbol{P}_r = \boldsymbol{K}_r(\boldsymbol{R}_r|\boldsymbol{t}_r)$ is

$$\boldsymbol{P_r} = \begin{bmatrix} f & 0 & p_r & -bf \\ 0 & f & q & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{3.11}$$

3.1.6 The Disparity

The term *disparity* was first introduced by Marr and Poggio (1976). It was used to describe the difference in location of corresponding features seen by the left and right eyes. This initial description is still used today in 2015 and has been extended to the difference in location of corresponding features seen by two cameras. The difference between the right and left x-coordinates of the projected points is called *disparity*. This difference is signed, and we chose the sign convention adopted in the

3D cinema (Mendiburu, 2009): the right camera image point minus the left camera image point. Given a 3D point in space $\boldsymbol{x} = (x, y, z)$ and a stereoscopic camera system defined by $\boldsymbol{P_l}$ and $\boldsymbol{P_r}$, the image disparity is

$$d(\boldsymbol{x}) = (\boldsymbol{P}_r\bar{\boldsymbol{x}})_x - (\boldsymbol{P}_l\bar{\boldsymbol{x}})_x \tag{3.12}$$

$$= p_r - \frac{bf}{z} - p_l. \tag{3.13}$$

Let us note that the disparity value only depends on the depth of the point $\boldsymbol{x}$, its third component $\boldsymbol{x}_z = z$. All 3D points at a plane parallel to the image plane at depth z have the same disparity. Moreover, points at depth $z = \infty$ have a finite disparity $d_0 = p_r - p_l$. We write the disparity as

$$d(\boldsymbol{x}) = d_0 - \frac{bf}{z}. \tag{3.14}$$

All points on a plane at distance $z = \frac{bf}{d_0}$ have disparity $d = 0$. This depth is known as the *convergence distance* of the stereo system and we note it H:

$$H = \frac{bf}{d_0}. \tag{3.15}$$

The convergence distance is usually adjusted by shifting the principal points p_l and p_r of one or both cameras, so that rays through the optical center and the image center intersect at a depth H. For practical purposes, the magnitude H is sometimes preferable over d_0, so we write the latter as a function of the first:

$$d_0 = \frac{fb}{H}. \tag{3.16}$$

Parallel Rectified Stereo Cameras The case where $d_0 = 0$ and $H = \infty$ is known as the *parallel rectified stereo camera*. The disparity is given by

$$d = -\frac{bf}{z}. \tag{3.17}$$

Moreover, by considering $b = 1$ and $f = 1$ and reversing the sign, we obtain the "standard" interpretation in computer vision of the *normalized disparity* (Okutomi and Kanade, 1993) as the inverse depth

$$d = \frac{1}{z}. \tag{3.18}$$

Assymetric and Symmetric Rectified Stereo Cameras It is sometimes practical to work with *convergent rectified stereo cameras* (Sec. 3.2 and Chapter 5). Their extrinsic parameters are defined by their position and translation in the world $(\boldsymbol{R}, \boldsymbol{c})$ as well as their *baseline* b and convergence distance H. Their focal length is

f and the difference between their principal points p_l and p_r is given by Eq. 3.16. If we choose $p_l = 0$ and $p_r = d_0$, the projection camera matrices P are

$$\boldsymbol{P}_l = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \text{and} \quad \boldsymbol{P}_r = \begin{bmatrix} f & 0 & \frac{bf}{H} & -bf \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{3.19}$$

This configuration is not symmetric as we chose the left camera to be on the origin of the world. Sometimes, in order to apply symmetry reasoning (Sec. 5.1), we use a symmetric parametrization of the *stereo cameras*:

$$\boldsymbol{P_l} = \begin{bmatrix} f & 0 & -\frac{bf}{2H} & \frac{bf}{2} \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \text{and} \quad \boldsymbol{P_r} = \begin{bmatrix} f & 0 & \frac{bf}{2H} & -\frac{bf}{2} \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{3.20}$$

Disparity units Let us note that the different representations of the disparity have different units. Let us assume z is in metric units. The disparity representation $d = \frac{1}{z}$ from Eq. 3.18 has inverse to metric units. The representation $d = d_0 - \frac{bf}{z}$ from Eq. 3.13 has pixel units, as b has metric units and p_l, p_r and f have pixel units. In some cases (Sec. 3.2) it is convenient to have disparity values as a fraction of the image width. Let w be the width of the image in pixel units. To obtain *normalized* disparity values without units we only need to normalize d_0 and the focal f with w:

$$d = \frac{d_0}{w} - \frac{f}{w}\frac{b}{z}. \tag{3.21}$$

3.1.7 3D to 3D Transformations: the Reconstruction Matrix

As we saw in Sec. 3.1.2, after a 3D to 2D projection we loose the depth information. In some cases it is important to project a 3D point into the image plane, but to keep the depth information. This is possible by using a full-rank 4x4 matrix $\tilde{\boldsymbol{P}}$, and not dropping the last row in the $\boldsymbol{P}$ matrix. As with the matrix $\boldsymbol{P}$, the extended matrix can be decomposed as

$$\tilde{\boldsymbol{P}} = \tilde{\boldsymbol{K}}\boldsymbol{E}, \tag{3.22}$$

where $\boldsymbol{E}$ is a 3D rigid-body (Euclidean) transformation and $\tilde{\boldsymbol{K}}$ is the full-rank calibration matrix. The matrix $\tilde{\boldsymbol{P}}$ is used to map directly from 3D homogeneous world coordinates $\tilde{\boldsymbol{q}}_W = (x_W, y_W, z_W, w_W)$ to image coordinates plus disparity, $\boldsymbol{x} = (x, y, 1, d)$, thus keeping the depth information in the projection process. We note

$$\tilde{\boldsymbol{x}} \propto \tilde{\boldsymbol{P}}\ \tilde{\boldsymbol{q}}_W, \tag{3.23}$$

where $\propto$ indicates equality up to scale. In this case the normalization is done with the *third* element of the vector to obtain the normalized form $\boldsymbol{x} = (x, y, 1, d)$. The 4x4 matrix $\tilde{\boldsymbol{P}}$ defines a *3D homography* of space.

In general, when using the 4 x 4 matrix $\tilde{\boldsymbol{P}}$, we have the freedom to choose the last row to whatever suits our purpose (Szeliski, 2010). The choice of the last row of $\tilde{\boldsymbol{P}}$ defines the mapping between depth and the last coordinate of the projected point d. For example, the "standard" *normalized disparity* as inverse depth $d = \frac{1}{z}$ (Okutomi and Kanade, 1993), is given by

$$\tilde{\boldsymbol{K}} = \begin{pmatrix} \boldsymbol{K} & \boldsymbol{0} \\ \boldsymbol{0}^\top & 1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{E} = \begin{pmatrix} \boldsymbol{R} & \boldsymbol{t} \\ \boldsymbol{0}^\top & 1 \end{pmatrix}. \tag{3.24}$$

We will use this disparity parametrization in Chapter 4.

When we work with a pair of rectified cameras we prefer to use the disparity defined by the difference of the two first rows of $\boldsymbol{P}_l$ and $\boldsymbol{P}_r$ (Eq. 3.13). In this case the obtained matrix is called *reconstruction matrix* (Devernay, 1997) and has the form

$$\tilde{\boldsymbol{P}} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{bf}{H} & -bf \end{pmatrix}. \tag{3.25}$$

The decomposition into the 3D rigid-body transformation and the full-rank calibration matrix is

$$\tilde{\boldsymbol{K}} = \begin{pmatrix} \boldsymbol{K} & \boldsymbol{0} \\ 0, 0, \frac{bf}{H} & -bf \end{pmatrix} \qquad \text{and} \qquad \boldsymbol{E} = \begin{pmatrix} \boldsymbol{R} & \boldsymbol{t} \\ \boldsymbol{0}^\top & 1 \end{pmatrix}. \tag{3.26}$$

The inverse of the reconstruction matrix The 4x4 matrix $\tilde{\boldsymbol{P}}$ is full-rank and therefore invertible. The inverse $\tilde{\boldsymbol{P}}^{-1}$ transforms points with disparity $\boldsymbol{x} = (x, y, 1, d)$ to 3D points in the world $\bar{q}_W = (x_W, y_W, z_W, 1)$. The relation between the disparity and the depth can be computed by inverting the disparity equations 3.13 and 3.18. If we use the disparity representing the pixel difference from Eq. 3.13 we obtain

$$z = \frac{bf}{(d_0 - d)}. \tag{3.27}$$

Or if we use the normalized version from Eq. 3.21 we obtain

$$z = \frac{bf}{(d_0 - wd)}. \tag{3.28}$$

The inverse of the reconstruction matrix will be used in Sec. 3.2.1 to determine the *perceived depth from stereopsis*.

3.2 Stereoscopic Filming: Acquisition and Projection

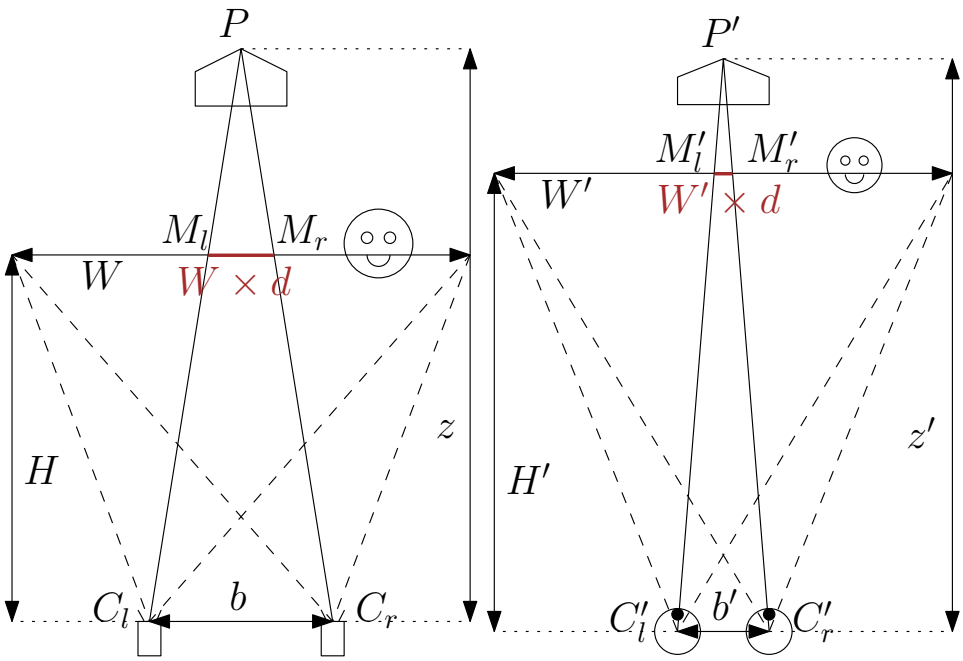
Stereoscopic movie-making process is a complex task involving mainly two stages: the acquisition and the projection. In the first stage the geometry is acquired with two cameras. In the projection stage, the two acquired images are projected onto the same screen in front of the spectator. An optical illusion is created: the 3D acquired scene is transformed into the 3D scene perceived by the spectator. The optical illusion is highly dependent on the acquisition and the projection parameters. Spottiswoode *et al.* (1952) wrote the first essay studying how the geometry is distorted by the "stereoscopic transmission" (i.e. acquisition and projection). Further studies (Woods *et al.*, 1993) extended these works and also computed spatial distortions of the perceived geometry. Masaoka *et al.* (2006) from the NHK conducted a similar study proposing a software tool allowing to predict the spatial distortions arising with a set of given acquisition and projection parameters. Devernay and Beardsley (2010) showed that a non-linear geometric 3D transformation exists between the 3D acquired scene and the 3D scene perceived by the spectator based on depth from stereopsis. This non-linear geometric 3D transformation can introduce 3D distortions creating visual fatigue and visual discomfort. As we saw in chapter 2, many depth cues play a role in the depth perception of the spectator. However, the study conducted by Held and Banks (2008), shows that the computation of the perceived depth from stereopsis provides a good prediction on the actual depth perceived by the audience. In the next section we present a geometrical study characterizing the 3D transformations and the 3D distortions of the perceived depth from stereopsis.

3.2.1 Perceived Depth from Stereopsis

Let us introduce the notation characterizing at the same time an *acquisition stereo system*, as well as a *projection stereo system*. In Fig. 3.5 we illustrate and summarize the notation. In the acquisition setup, the distance between the optical centers of the camera b is called *baseline*, *interocular* or *interaxial*. The cameras *convergence distance* is H (see Sec. 3.1.6), and we name the parallel plane to the images at distance H the *convergence plane*. The intersection of the camera visibility frustums with the convergence plane defines the convergence window, and we note its width W. With an abuse of notation, W is usually referred to as the *convergence plane width*. In the projection setup, the distance between the eyes of the spectator is b'. The distance between the spectator and the screen where the images are projected is H' and the width of the screen is W'. For the rest of our work we assume all parameters b, b', H, H', W and W' to be greater than 0.

Let us use the asymmetric rectified configuration of Sec. 3.1.6. With this parametrization, the focal of the cameras f and the acquisition convergence disparity d_0, in pixel units, are given by the relations

$$f = w\frac{H}{W} \quad \text{and} \quad d_0 = \frac{fb}{H}. \tag{3.29}$$



Symbol	Acquisition	Projection
$\mathbf{C}_l$, $\mathbf{C}_r$	camera optical center	eye optical center
$\mathbf{P}$	physical point of the scene	perceived 3D point
$\mathbf{M}_l$, $\mathbf{M}_r$	image points of $\mathbf{P}$	screen points
b	baseline	humain eye distance
H	convergence distance	screen distance
W	convergence plane size	screen size
z	real depth	perceived depth
d	left-right disparity (as a fraction of W)	

Fig. 3.5: *Parameters describing the shooting geometry and the movie theater configuration (reproduced from Devernay and Beardsley (2010)).*

Analogously, the focal of the spectator f' and the projection convergence disparity d'_0, in pixel units, are given by the relations

$$f' = w\frac{H'}{W'} \quad \text{and} \quad d'_0 = \frac{f'b'}{H'}. \tag{3.30}$$

By plugging the relations from Eq. 3.29 into Eq. 3.21 we obtain the normalized disparity

$$d = \frac{b}{W}\frac{(z-H)}{z}. \tag{3.31}$$

Inversely, given a normalized disparity $d' = d$ and the projection parameters b', H' and W', and plugging the relations from Eq. 3.30 into Eq. 3.28, we obtain the

perceived depth from stereopsis z'

$$z' = \frac{H'}{1 - \frac{W'}{b'}d}. \tag{3.32}$$

The relationship between the true depth in the 3D scene and the perceived depth from stereopsis in the projection room can be written by combining Eq. 3.31 and Eq. 3.32, as proposed by Devernay and Beardsley (2010). The obtained relationship is given by

$$z' = \frac{H'}{1 - \frac{W'}{b'}(\frac{b}{W}\frac{z-H}{z})}. \tag{3.33}$$

In some cases it will be more convenient to re-write this expression as:

$$z' = \frac{zb'H'W}{z(b'W - bW') + bHW'}. \tag{3.34}$$

3.2.1.1 Canonical Setup

The shooting configuration

$$b'W = bW' \quad \text{or} \quad \frac{b}{b'} = \frac{W}{W'} \tag{3.35}$$

creates a linear relation between z' and z:

$$z' = z\frac{H'}{H}. \tag{3.36}$$

This configuration is known as the *Canonical setup* (Devernay and Beardsley, 2010).

Furthermore, by choosing $\frac{H'}{H} = 1$ the perceived depth z' is equal to z. Although this configuration may seem interesting, we will see (Sec. 3.2.4) that it may introduce important 3D distortions of the perceived scene.

3.2.1.2 Homothetic Setup

A more convenient configuration is

$$\frac{b'}{b} = \frac{H'}{H} = \frac{W'}{W}, \tag{3.37}$$

known as the *homothetic configuration* (Devernay and Beardsley, 2010).

3.2.2 Perceived Position from Stereopsis

In our work we are not only interested in the perceived depth from stereopsis, but also in the general 3D perceived position from stereopsis. As we will see,

some phenomena responsible for visual fatigue or visual discomfort depend not only on the perceived depth, but also on the perceived position. To model the 3D transformation we use the reconstruction matrix mapping 3D points to image points plus disparity, and its inverse, mapping image points plus disparity to 3D points (Sec. 3.1.7).

The projection of the 3D scene points into image point plus disparity is given by the filming parameters H, W, b. Let us write the filming reconstruction matrix $\tilde{\boldsymbol{P}}_f$ from Eq. 3.25 using the acquisition parameters. In this case we consider a normalized focal without units $\frac{H}{W}$, so that image coordinates are normalized. A 3D point in the scene $\boldsymbol{x} = (x, y, z)$ is projected into a normalized image coordinate plus disparity $\boldsymbol{u} = (u, v, 1, d)$ with $\tilde{\boldsymbol{P}}_f \bar{\boldsymbol{x}} = \tilde{\boldsymbol{u}}$ where

$$\tilde{\boldsymbol{P}}_f = \begin{bmatrix} \frac{H}{W} & 0 & 0 & 0 \\ 0 & \frac{H}{W} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{b}{W} & -b\frac{H}{W} \end{bmatrix}. \tag{3.38}$$

The image point is obtained by normalizing with the *third* element of $\tilde{\boldsymbol{u}}$:

$$\boldsymbol{u} = \begin{bmatrix} \frac{x}{z}\frac{H}{W} \\ \frac{y}{z}\frac{H}{W} \\ 1 \\ \frac{b}{W}\frac{(z-H)}{z} \end{bmatrix}. \tag{3.39}$$

The reconstruction matrix of the projection system $\tilde{\boldsymbol{P}}_p$ can be obtained by replacing b, H and W with b', H' and W' in $\tilde{\boldsymbol{P}}_f$:

$$\tilde{\boldsymbol{P}}_p = \begin{bmatrix} \frac{H'}{W'} & 0 & 0 & 0 \\ 0 & \frac{H'}{W'} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{b'}{W'} & -b'\frac{H'}{W'} \end{bmatrix}. \tag{3.40}$$

In this case we are interested in the inverse of $\tilde{\boldsymbol{P}}_p$, mapping a normalized image point with disparity $\boldsymbol{u} = (u, v, 1, d)$ into a 3D point $\boldsymbol{x'}$. The determinant of the matrix $\tilde{\boldsymbol{P}}_p$ is $-b'(\frac{H'}{W'})^3$, which is only zero if either H' or b' are zero. As we assumed that all parameters $b', H', W' > 0$, the matrix $\tilde{\boldsymbol{P}}_p$ is invertible and its inverse is

$$\tilde{\boldsymbol{P}}_p^{-1} = \begin{bmatrix} \frac{W'}{H'} & 0 & 0 & 0 \\ 0 & \frac{W'}{H'} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{H'} & -\frac{W'}{b'H'} \end{bmatrix}. \tag{3.41}$$

The 3D homography $\tilde{\boldsymbol{H}}$ transforming a 3D point in the acquired scene $\boldsymbol{x}$ into a 3D point in the perceived scene $\boldsymbol{x}'$ is given by the product of $\tilde{\boldsymbol{P}}_p^{-1}$ with $\tilde{\boldsymbol{P}}_f$:

$$\tilde{\boldsymbol{H}} = \begin{bmatrix} \frac{HW'}{WH'} & 0 & 0 & 0 \\ 0 & \frac{HW'}{WH'} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{H'} - \frac{bW'}{b'H'W} & \frac{bHW'}{b'H'W} \end{bmatrix}. \tag{3.42}$$

The perceived 3D point in homogeneous coordinates is

$$\tilde{\boldsymbol{x}}' = \begin{bmatrix} x\frac{HW'}{WH'} \\ y\frac{HW'}{WH'} \\ z \\ \frac{z(b'W-bW')+bHW'}{b'H'W} \end{bmatrix}. \tag{3.43}$$

By normalizing with the fourth component we obtain the coordinates of the perceived position from stereopsis $\boldsymbol{x}' = (x', y', z')$, with

$$x' = x\frac{b'HW'}{z(b'W - bW') + bHW'}, \tag{3.44}$$

$$y' = y\frac{b'HW'}{z(b'W - bW') + bHW'}, \tag{3.45}$$

and

$$z' = \frac{zb'H'W}{z(b'W - bW') + bHW'}. \tag{3.46}$$

The perceived depth is, as expected, equal to Eq. 3.34.

Now that we have written the 3D homography between the filmed 3D scene and the perceived 3D from stereopsis, let us mathematically characterize the phenomena responsible for visual fatigue and visual discomfort. When possible, we deduce the shooting baseline b avoiding such effects.

3.2.3 Ocular Divergence Limits

Ocular divergence happens when both eyes look at the screen with a negative angle between them. Both viewing rays intersect behind the spectator, as we illustrate in Fig. 2.2. The mathematical condition of eye divergence is then $z' < 0$.

Lets us recall Eq. 3.34 :

$$z' = \frac{zb'H'W}{z(b'W - bW') + bHW'}. \tag{3.47}$$

The numerator can not be negative because z, b', W, H' are all positive. The

denominator can be negative

$$bHW' + z(b'W - bW') < 0. \tag{3.48}$$

If $b'W - bW' \geq 0$ there is no divergence. If $b'W - bW' = 0$ then $z \to +\infty \implies z' \to +\infty$. The equality establishes the biggest non-divergence baseline:

$$b = b'\frac{W}{W'}. \tag{3.49}$$

This configuration is the *Canonical Setup* from Eq. 3.35.

3.2.3.1 Divergence Depth

If $b'W - W'b < 0$, then when

$$z \to -\frac{bHW'}{(b'W - bW')} \implies z' \to +\infty. \tag{3.50}$$

Elements at $z > -\frac{bHW'}{(b'W-bW')}$ cause eye divergence in the projection room. We note this magnitude as the divergence limit

$$z_{\text{Div}} = -\frac{bHW'}{(b'W - bW')}. \tag{3.51}$$

3.2.3.2 Perceived Depth of Infinity

If $b'W - bW' > 0$, then eye divergence does not happen and

$$z \to +\infty \implies z' \to \frac{b'H'W}{(b'W - bW')}. \tag{3.52}$$

Elements at $z = +\infty$ are transformed into the finite location

$$z'(\infty) = \frac{b'H'W}{(b'W - bW')}. \tag{3.53}$$

Solving Eq. 3.53 for b we obtain the baseline mapping $z = +\infty$ to the desired $z'(\infty)$. We obtain

$$b = b'\frac{W}{W'}\frac{z'(\infty) - H'}{z'(\infty)}. \tag{3.54}$$

This baseline is important to avoid the vergence-accommodation conflict (see Sec. 2.2.1). Perceived 3D points farther than this limit may cause visual fatigue.

Example: a small display Let us assume the projection parameters are fixed (b', H', W'), as well as the shooting convergence plane width and distance (H, W). When looking at a small display, e.g. a mobile phone or tablet, at a distance of

$H' = \frac{1}{3}$m (≈ 0.33 m), elements perceived farther than $z' = \frac{1}{2.25}$m (≈ 0.44 m) cause visual discomfort (Banks *et al.*, 2013). When creating stereoscopic content for a small display of width $W' = 0.20$m, elements at infinity should not be projected farther than $z' \approx 0.44$ m. By substituting in Eq. 3.54 we obtain

$$b \approx 0.065\ W\ \frac{(0.44 - 0.33)}{0.44\ \times\ 0.2} = 0.08125W. \tag{3.55}$$

Note the important magnification (400%) compared to the baseline obtained with the *Canonical Setup* ($bW' = b'W$):

$$b \approx \frac{0.065}{0.2}\ W = 0.325\ W. \tag{3.56}$$

A comment on diverging configurations A human is capable to perform ocular divergence within a small range ($0.5 - 1°$)(Shibata *et al.*, 2011). Some stereographers take advantage of this fact and use a divergent configuration ($b > b'\frac{W}{W'}$) to map the farthest object in the scene, *farther away* than infinity in the projection room. Although there is no substantial difference for those far objects, as they are still perceived at infinity, this configuration with a bigger baseline allows to increase the *roundness factor* around the depth of the screen (Eq. 3.65). In the next section (3.2.4) we introduce the *roundness factor*.

3.2.4 Roundness Factor

The scene distortions in the perceived scene come from different scene magnifications in the fronto-parallel directions (width and height), and in the depth direction. Spottiswoode *et al.* (1952) defined the *shape ratio* as the ratio between depth magnification and width magnification. Mendiburu (2009) and Devernay and Beardsley (2010) use the term *roundness factor*. The roundness factor at a depth z is defined as the ratio between the depth variation in the perceived space with respect to the scene depth ($\frac{\partial z'}{\partial z}$) and the apparent size variation with respect to space ($\frac{\partial x'}{\partial x}$, or $\frac{\partial y'}{\partial y}$):

$$\rho(z) = \frac{\frac{\partial z'}{\partial z}}{\frac{\partial x'}{\partial x}}(z). \tag{3.57}$$

The partial derivatives of the perceived position with respect to the x and y coordinates of the acquired position (Eqs. 3.44 and 3.45) are:

$$\frac{\partial x'}{\partial x}(z) = \frac{b'HW'}{z(b'W - bW') + bHW'}, \tag{3.58}$$

$$\frac{\partial y'}{\partial y}(z) = \frac{b'HW'}{z(b'W - bW') + bHW'}, \tag{3.59}$$

$$\frac{\partial x'}{\partial y}(z) = 0 \quad \text{and} \quad \frac{\partial y'}{\partial x}(z) = 0. \tag{3.60}$$

Note that for scene elements at the convergence distance $z = H$, their apparent size ratio simplifies to $\frac{W'}{W}$.

The partial derivative of the perceived depth with respect to z (Eq. 3.34) is

$$\frac{\partial z'}{\partial z} = \frac{bb'HH'WW'}{(z(b'W - bW') + bHW')^2}. \tag{3.61}$$

Plugging $\frac{\partial x'}{\partial x}$ from Eq. 3.58 and $\frac{\partial z'}{\partial z}$ from Eq. 3.61 into Eq. 3.57 we obtain the expression of the roundness of an element at depth z:

$$\rho(z) = \frac{bH'W}{z(b'W - bW') + bHW'}. \tag{3.62}$$

In Fig. 3.6 we illustrate the different values of the roundness factor.

Interesting configurations Let us analyze some interesting cases. In the *Canonical Setup* 3.35 the roundness is constant for all depths:

$$\rho(z) = \frac{H'}{H}\frac{W}{W'} = \frac{H'}{H}\frac{b}{b'}. \tag{3.63}$$

In the *Homothetic Setup* (Eq. 3.37) the roundness is 1 for all depths:

$$\rho(z) = 1. \tag{3.64}$$

Independently of the chosen configuration, at the screen plane depth $z = H$, the roundness of the perceived depth is independent of the screen width W':

$$\rho(H) = \frac{b}{b'}\frac{H'}{H}. \tag{3.65}$$

3.2.4.1 Cardboard Effect

The cardboard effect arises when the roundness of a scene element is smaller than 0.3 (Mendiburu, 2009). Elements of the scene are perceived in depth, but they are themselves flat, *as if they were drawn on a cutout cardboard* (Sec. 2.1.4). Let us rewrite the roundness equation 3.62 by writing z as a fraction of H: $z = \lambda H$. Then we obtain

$$\rho(\lambda) = \frac{H'}{H}\frac{bW}{(\lambda(b'W - bW') - bW')}. \tag{3.66}$$

With this factorization, the term $\frac{H'}{H}$ can be seen as an amplitude coefficient. If we keep b and W constant, but we increase H (the distance of the cameras to the convergence plane), the roundness gets smaller. This is known to be the *cardboard effect*, introduced for example by the use of long focal lengths (Sec. 3.5).

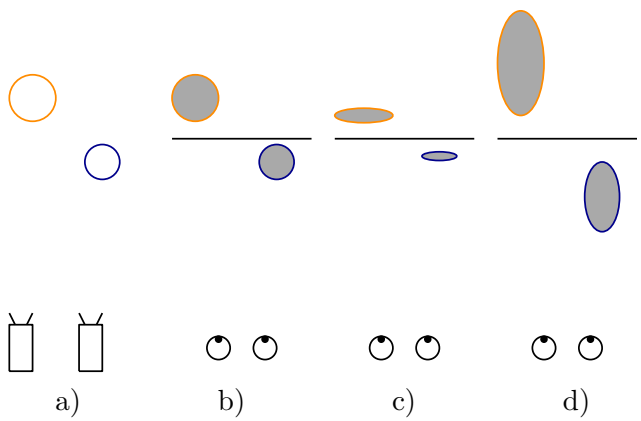


Fig. 3.6: *Scheme of the roundness factor. a) Shooting two spheres. b) obtaining a roundness factor equal to 1. c) obtaining a roundness factor smaller than 1. d) obtaining a roundness factor bigger than 1.*

3.2.5 Relative Perceived Size of Objects

The 3D transformation of the acquired scene into the perceived scene, may not only modify the perceived depth of the elements, but also its perceived size. It is known that a perspective transformation makes big objects being far away to appear small on the screen. Two similar objects with different sizes on the screen lead the audience to think they are far apart. This is known as the relative size depth cue (see Sec. 2.1.1). If this depth cue is inconsistent with the perceived depth from stereopsis, it may introduce a perception distortion called *puppet theater effect* (Sec. 2.1.4).

Yamanoue *et al.* (2006) propose a geometric predictor E_p of the *puppet-theater effect*, based on the depth perception from stereopsis. They first define the apparent magnification of an object $M(z)$ as the ratio between its actual size and the perceived size. An object of size w is seen in the projection room as having a size of $w' = M(z)w$. In our terms we write this magnification factor as

$$M(z) = \frac{\partial x'}{\partial x}(z). \tag{3.67}$$

Then they define the predicted amount of puppet-theater effect E_p, as the ratio between the magnification factors at a foreground depth ($M(z_f)$) and the magnification factor at a background depth ($M(z_b)$). With our notation we write this magnitude as

$$E_p(z_f, z_b) = \frac{\frac{\partial x'}{\partial x}(z_f)}{\frac{\partial x'}{\partial x}(z_b)}. \tag{3.68}$$

If the predictor value E_p is close to one, there is no puppet-theater effect, while if the predictor value is smaller, the 3D projected scene may create the puppet-theater effect. In their subjective test they found out that a subject of interest appears to

have its *normal size* when $E_p \in (0.75, 1.25)$. Whereas outside this range, subjects reported a distorted scale of the scene objects. One of their straightforward claims is that, if the magnification factor is independent of z, e.g. in the *Canonical setup*, then there is no puppet-theater effect.

Similarly Devernay and Duchêne (2010) define the *image scale ratio* σ', which is how much an object placed at depth z seems to be enlarged with respect to objects in the convergence plane ($z = H$). The magnification of objects at the convergence plane is $\frac{W'}{W}$ and thus

$$\sigma'(z) = \frac{W'}{W} \frac{1}{\frac{\partial x'}{\partial x}(z)}. \tag{3.69}$$

To obtain a one parameter expression of the *puppet-theater effect* predictor we can chose a reference object to be at $z_f = H$. Then for an object at any depth z, (greater or less than H) we can compute the *puppet-theater effect* predictor E_p. Eq. 3.68 becomes

$$E_p(H, z) = \frac{1}{\sigma'(z)}, \tag{3.70}$$

and using Eq. 3.58 we obtain

$$E_p(z) = \frac{z(b'W - bW') + bHW'}{b'HW}. \tag{3.71}$$

3.2.6 Changing the Projection Geometry

It is well known that projecting a sterescopic movie on different screens with different screen sizes and different viewing distances produces different depth perceptions (Spottiswoode *et al.*, 1952; Lipton, 1982; Mendiburu, 2009; Devernay and Beardsley, 2010; Chauvier *et al.*, 2010). To control and to adapt the disparity to the viewing situation is of central importance to the widespread adoption of stereoscopic 3D (Sun and Holliman, 2009).

A stereoscopic film is shot for a given projection configuration, usually named *target screen*. Displaying it in a different projection room, with a different screen width from the original target screen, creates distortions of the perceived depth from stereopsis. For example, when projecting a film in a movie theater and on a 3D television the perceived depth will be different. If the film is projected on a bigger screen than the target screen, it may even cause eye divergence, as on-screen disparities are scaled proportionally to the scale of the screen width. When the disparities are bigger than the human interocular, ocular divergence occurs (Sec. 3.2.3). In Fig. 3.7 we illustrate the modification of the perceived depth from stereopsis when the projection screen is scaled. Note that a change in W' affects the perceived depth from stereopsis (Eq. 3.33), the roundness (Eq. 3.62), the ocular divergence limit (Eq. 3.51) and the relative perceived size of objects (Eq. 3.71).

Changing the viewing distance H' of the spectator also modifies the perceived depth from stereopsis (Eq. 3.33) and the roundness (Eq. 3.62). However it does neither affect the ocular divergence limit (Eq. 3.51) or the relative perceived size of

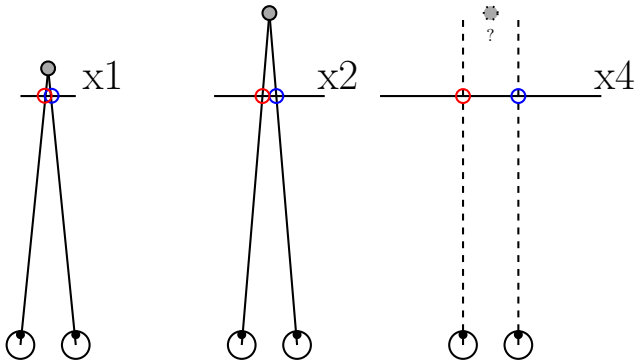


Fig. 3.7: *Impact of the screen size to the depth perception. The perceived depth from stereopsis changes when the width of the screen changes. If the on-screen disparity is bigger than the human interaxial, it may cause eye divergence.*

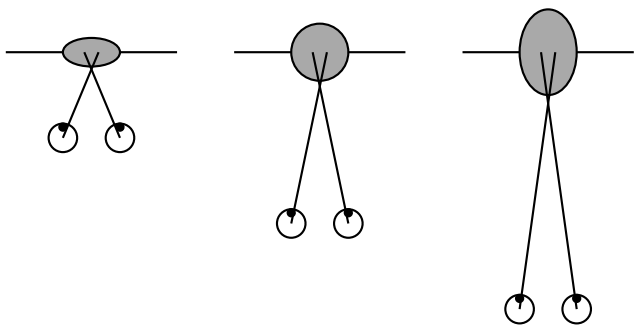


Fig. 3.8: *Impact of the distance to the screen (H') in the depth perception. The perceived depth from stereopsis changes when the distance to the screen changes. The roundness of the object scales linearly with the spectator distance to the screen (Eq. 3.62).*

objects (Eq. 3.71). In Fig. 3.8 we illustrate the impact of the viewing distance H' to the depth perception, and in particular, to the roundness factor.

3.2.7 The Ideal Viewing Distance

As we just saw, the perception of depth may significantly vary depending on the projection configuration H' and W'. However, it seems reasonable to assume that there is a relation between both. If the screen is bigger, the spectator sits farther away, whereas if the screen is small, the spectator sits (or stands) closer to the screen. This idea was already stated by Spottiswoode *et al.* (1952), claiming that the standard distance from spectator to screen should be from $2W'$ to $2.5W'$. Recent recommendations from SMPTE (2015) and THX (2015) establish the *acceptable* viewing distances from a screen by fixing a range of viewing angles. For example, the *SMPTE STANDARD 196M-2003* defines the maximal recommended horizontal viewing angle of 30° (SMPTE, 2003), whereas the *THX certified screen placement* states the maximal acceptable horizontal viewing angle of 36° for cinema theaters (THX, 2015a). *THX* recommendations also state that the best viewing distance for an HDTV setup is defined by a viewing angle of 40° (THX, 2015b). A viewing

angle of 36° establishes the relationship between H' and W'

$$1.6W' = H'. \tag{3.72}$$

Although there is a variability on the *ideal viewing distance* and no unanimous decision can be found across the different recommendations, it is possible to extrapolate a rough dependency between W' and H'. If we follow the THX recomendations for big screens, and take into account the nearest distance at which we can properly focus (around 0.33m), one could define the viewing distance as a function of the screen width as follows:

$$H'(W') = \begin{cases} 1.6\ W' & \text{if } A < W' \\ \text{some smooth function} & \text{if } B \leq W' \leq A \\ 0.33 & \text{if } W' < B. \end{cases} \tag{3.73}$$

The parameter A is somewhere around 1 to 2m where the preferred viewing distance may be bigger than the proposed 1.6 W'. To view a 1m width TV at a 1.6m distance seems way to near. The parameter B is somewhere around a tablet device width ($20 - 30$cm), where one does not hand-hold the device anymore and sets it on a table to sit farther away than 33cm.

An interesting study pointing in the same direction (Banks *et al.*, 2014) shows that the preferred viewing distance of a spectator when looking at an image of width W, is around 1.42 W. This study was performed with images with sizes from 15cm to 1m. The preferred viewing distance linearly scales with the width of the image. Most interestingly, the preferred viewing distance corresponds to the field of view of a 50mm focal length. In the cinema, this focal length is known as providing the *most natural* perception of the scene.

In our work we use the hypothesis that a function $H'(W')$ exists. Although the function might not be exact, it provides a mean to reduce the 2 dimensional space (H', W') into a one dimensional manifold parametrized by W': $(W', H'(W'))$.

3.3 The Virtual Projection Room

Understanding the 3D distortions introduced by the 3D transformation between the acquired scene and the perceived scene is not straightforward, because of the non-linear transformation from Eq. 3.33. While top view schemes illustrating the different distortions in very schematic configurations are helpful (Figs. 3.7, 3.8, 3.6 or 2.2), it is difficult to *see* how a generic 3D scene is distorted when acquired and projected with a set of parameters b, H, W and b', H', W'. Masaoka *et al.* (2006) proposed a visualization tool to explore the spatial distortions, providing a top view of the perceived depth from stereopsis (see Fig. 3.9). The acquisition and projection parameters can be adjusted and the 3D distortions are displayed.

We propose to go further and create a *virtual projection room* in a 3D environment (Blender, 2015), allowing to *see* the 3D deformations of a generic 3D scene when

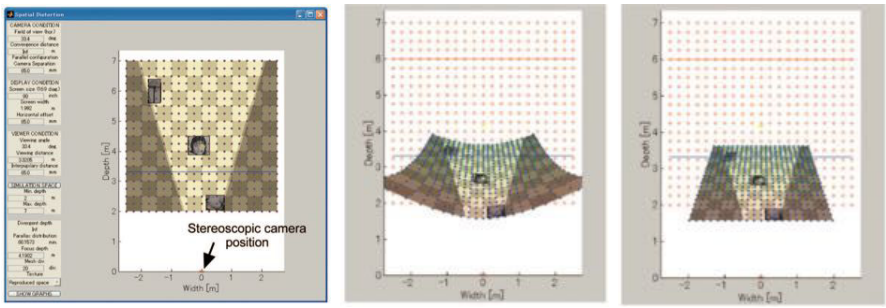


Fig. 3.9: *Illustrations reproduced from Masaoka et al. (2006). The non-linear 3D distortions between the acquired scene and the perceived scene are shown as a top view of the projection room. Elements in the scene are characterized by 2d pictures.*

acquired and projected with a set of parameters b, H, W and b', H', W'. Compared with previous work we provide an interactive 3D view of the distortions, as scene elements can be animated, and the acquisition camera parameters adjusted over time.

The *virtual projection room* is a new visualization tool allowing to interactively *see* the perceived depth from stereopsis by the spectator. On one side we have a 3D model of the scene and the acquisition cameras. On the other side we have the spectator on his couch at home (or in the cinema) looking at the projected images. Both the acquisition geometry (b, H, W) as well as the projection geometry (b', H', W') can be adjusted at will. Fig. 3.10 and 3.11 illustrate the acquisition and projection stages. We use the *virtual projection room* to illustrate with a series of figures the 3D distortions described in Sec. 3.2. We use a "Toy Scene" consisting of a woman and two spheres and a stereoscopic pair of cameras acquiring the scene. In Fig. 3.10 we illustrate the acquisition of the scene. In Fig. 3.11 we illustrate the perceived depth from stereopsis as we project the acquired images in the virtual projection room describing a home cinema. In Fig. 3.12 we illustrate the 3D deformations arising when shooting with a deviating from the *Canonical Setup* (Sec. 3.2.1.1), i.e. $b < b'\frac{W}{W'}$ and $b > b'\frac{W}{W'}$. In Fig. 3.13 we illustrate the 3D deformations arising when projecting the images on different screen sizes and viewing distances (Sec. 3.2.6). In Fig. 3.14 we illustrate the *cardboard effect* (Sec. 3.2.4.1). In Fig. 3.15 we illustrate the *puppet theater effect* (Sec. 3.2.5).

The *virtual projection room* is integrated into the stereoscopic shooting simulator *Dynamic Stereoscopic Previz* (Pujades *et al.*, 2014), that we briefly present in Appendix A. The *Dynamic Stereoscopic Previz* (DSP) is a video game where the goal is to shoot a stereoscopic film. The user first models and animates a 3D scene using the Blender Game Engine. Then the user places a stereoscopic rig in the scene and adjust the shooting parameters at will (b, H, W). The user also sets the parameters of the *virtual projection room* (b', H', W'), and sees how the acquired images are perceived by the spectator. The *virtual projection room* is updated in real-time, as the user changes the shooting parameters. The shooting simulator was tested during the actual production of the short stereoscopic movie "Endless Night". In Appendix A we illustrate one shot of the movie with the DSP in action.

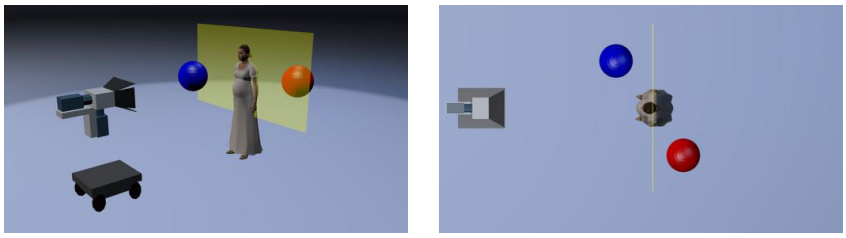


Fig. 3.10: *Acquisition of the Toy Scene. The woman is at 2.5m of the camera, and the woman's shoulders are 0.5m wide. The spheres have a diameter of 0.5m. The blue one is 0.5m in front of the woman, and the red one 0.5m behind. The acquisition parameters are: the baseline* $b = 65mm$*, the convergence distance* $H = 2.5m$ *(the depth of the woman), and the convergence window width* $W = 2.5m$*. A yellow window shows the convergence window, to help the operator validate the parameters. Left: a perspective view. Right: a top orthogonal view.*

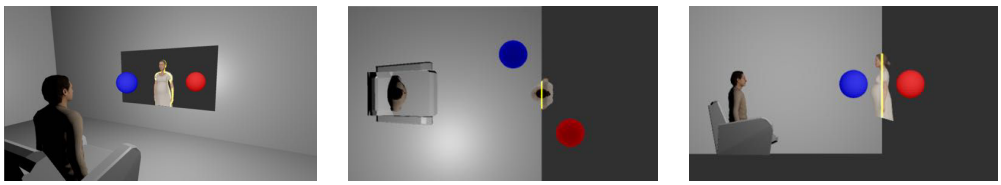


Fig. 3.11: *Projection of the Toy Scene in the virtual projection room. The acquisition configuration is* $b = 65mm$*,* $H = 2.5m$ *and* $W = 2.5m$ *(Fig. 3.10). The projection parameters are* $b' = 65mm$ *(human interocular of the spectator),* $H' = 2.5m$ *(distance of spectator to screen), and* $W' = 2.5m$ *(width of the screen). Because the virtual projection room configuration matches the acquisition configuration, no 3D distortion is introduced. The 3D transformation is the Identity transformation. Left: perspective view. Center: top orthogonal view. Right: lateral orthogonal view.*

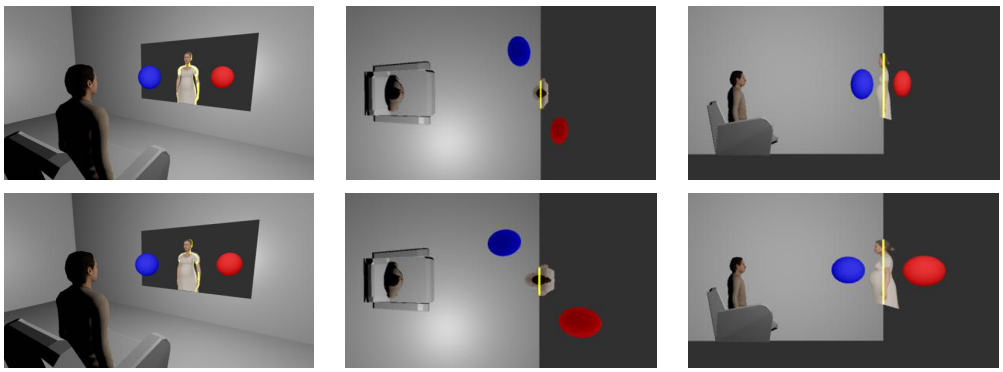


Fig. 3.12: *3D distortions appearing with the modification of the acquisition baseline* b*. The projection configuration is* $b' = 65mm$*,* $H' = 2.5m$ *and* $W' = 2.5m$*. Top row: hypo-stereo configuration with* $b = 45mm$*. Bottom row: hyper-stereo configuration with* $b = 85mm$*. Left: perspective view. Center: top orthogonal view. Right: lateral orthogonal view.*

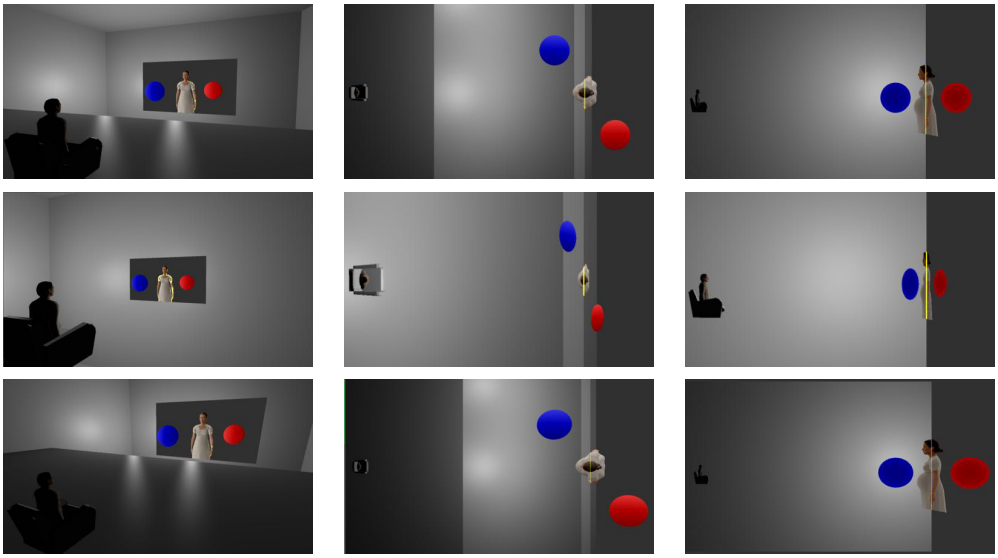


Fig. 3.13: *Changing the projection geometry. Acquisition configuration* $b = 16.25mm$, $H = 3.75m$ *and* $W = 2.5m$ *intended for the target screen* $b' = 65mm$, $H' = 15m$ *and* $W' = 10m$. *Top row: viewing the acquired images on the target screen. Center row: viewing the acquired images on a smaller screen* $b' = 65mm$, $H' = 7.5m$ *and* $W' = 5m$. *Bottom row: viewing the acquired images on a bigger screen* $b' = 65mm$, $H' = 18m$ *and* $W' = 12m$. *In the left column (a perspective view) it is difficult to see how the different projection configurations affect the perceived depth. However, in the center and right columns (top and lateral orthogonal views) we can see how the depth perception is affected. When viewing the images in the target screen (top row), spheres are perceived as spheres. Reducing the width of the screen (middle row) reduces and distorts the depth perception: spheres look flatter. Increasing the width of the screen (bottom row) increases and distorts the depth perception: the spheres are no longer spheres. Moreover, ocular divergence may appear.*

3.4 Adapting the Content to the Width of the Screen

As we saw in Sec. 3.2.6, projecting a stereoscopic film with a different projection configuration from the target configuration, modifies the depth perception and may introduce geometric distortions. In order to avoid those distortions, novel view synthesis techniques propose to adapt the content of the images. The literature in this domain is extensive. We briefly review the most popular methods.

Methods adapting the content to the width of the screen usually involve three steps. First the disparity between the left and right view is computed. The obtained disparity map might be dense, i.e. every pixel of the image has a depth value, or sparse, i.e. only a set of image correspondences are computed. The second step is the computation of a disparity mapping function, usually noted with $\phi(d) : \mathbb{R} \rightarrow \mathbb{R}$, converting the disparity values from the original stereo pair into the desired disparity values for the novel view. The last step is to render a novel view, so that the final stereoscopic pair has the mapped disparity values.

Disparity Computation Dense disparity maps are computed with stereo methods (Scharstein and Szeliski, 2002), and the computation of disparity maps

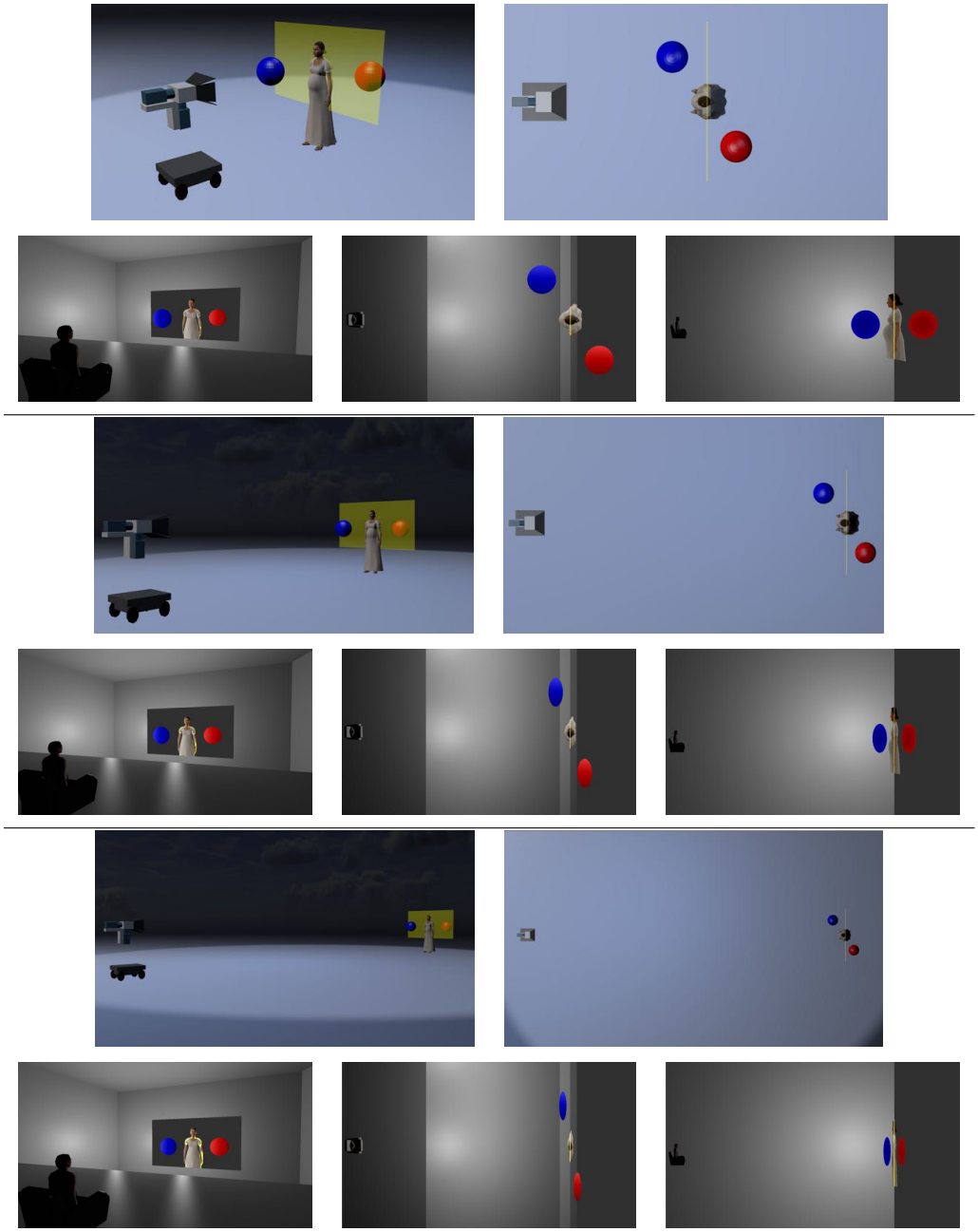


Fig. 3.14: *The cardboard effect. Projection is always done with the same configuration:* $b' = 65mm$, $H' = 15m$ *and* $W' = 10m$. *First and second rows: acquisition with Homothetic Setup* $b = 16.25mm$, $H = 3.75m$, $W = 2.5m$. *The woman and the spheres are perceived without distortion. Third and fourth rows: The camera moves backwards and changes the focal length to obtain the same convergence window. Acquisition parameters* $b = 16.25mm$, $H = 7.5m$, $W = 2.5m$. *The roundness of the woman and the spheres is divided by 2. They are not "round" anymore. Fifth and sixth rows: The camera moves backwards and changes the focal length to obtain the same convergence window. Acquisition parameters:* $b = 16.25mm$, $H = 15m$ *and* $W = 2.5m$. *The roundness of the woman and the spheres is divided by 4. The cardboard effect increases, the woman and the spheres look "flatter" to the spectator.*

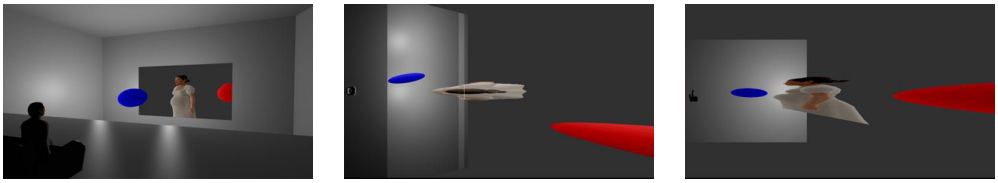


Fig. 3.15: *The puppet theater effect. The acquisition configuration is $b = 77mm$, $H = 2.5m$ and $W = 2.5m$. The projection configuration is $b' = 77mm$, $H' = 15m$ and $W' = 10m$. The blue sphere and the red sphere have exactly the same size in the acquired scene, and a very similar size on the acquired images, as seen in the left perspective view of the virtual projection room. However, as shown in the top and lateral orthogonal views, the blue sphere is perceived close to the spectator, and the red one far away. Because of the relative size of objects, the spectator perceives the blue sphere as normal size, whereas the red sphere is perceived as "huge".*

is still a very active research topic. In general, methods compute the cost of a pixel to have a given disparity, and find the disparity map minimizing the global cost of the image. Examples are, semi global block matching (Hirschmüller, 2008) or Sinha *et al.* (2014).

Sparse feature correspondences can be obtained with well established standard techniques (Baker and Matthews, 2004; Lowe, 2004). Those techniques do not provide features in large textureless images regions and may contain false matches, known as outliers. To counter those drawbacks, Lang *et al.* (2010) propose to exploit downsampled dense correspondence information using optical flow (Werlberger *et al.*, 2009), and to automatically filter the outliers with SCRAMSAC (Sattler *et al.*, 2009), an improvement of the well known RANSAC method (Fischler and Bolles, 1981).

Disparity Mapping Functions The goal of a disparity mapping function $\phi(d)$ is to reduce the distortions in the 3D transformation between the acquired scene and the perceived scene (Sec. 3.2). By a clever modification of the disparity of the projected stereoscopic pair, the depth distortions can be reduced. The simplest form of disparity mapping function is a linear mapping, like for example the one proposed by Kim *et al.* (2008). A linear mapping of the disparity corresponds to a view interpolation between the two original views. The disparity mapping function can also be non linear, either defined as a single function (Devernay and Duchêne, 2010) or as a combination of disparity mapping operators (Lang *et al.*, 2010). In Sec. 3.4.2 we mathematically study their impact in the perceived depth. In Fig. 3.16 we illustrate the shape of a non-linear disparity mapping function.

Rendering Once the disparity is modified, the novel view synthesis problem basically reduces to a view interpolation problem. In the literature of novel view synthesis to adapt stereoscopic content to the viewing conditions, we can distinguish two different types of methods. A first group of methods use *dense disparity maps warps* to render the target views, and a second group of methods use *content aware warps*. We briefly discuss their advantages and drawbacks.

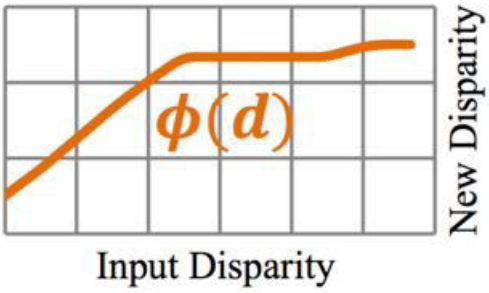


Fig. 3.16: *An example of a disparity mapping function $\phi(d)$ from Lang* et al. *(2010). In the first part of the function, a linear mapping preserves the depth. After a certain depth, the function is almost flat, compressing a large depth range into a smaller range of depth values. Scene elements that were far away are pulled forward in depth.*

Dense geometry warps Methods using a dense geometry belong to the family of depth image-based rendering (DIBR) methods. A detailed scheme of how DIBR methods work can be found in Zinger *et al.* (2010). The basic idea is to generate a virtual viewpoint using texture and depth information of the original images. Artifacts are usually removed by post-processing the projected images. These images are then blended together and the remaining disocclusions are filled in by inpainting techniques (Oh *et al.*, 2009; Jantet *et al.*, 2011).

The main drawback of DIBR methods, is that any error in the disparity estimation will generate artifacts in the final generated views. These can appear only on one view and disrupt the 2D image quality, or appear in both views, thus creating 3D artifacts, i.e. floating bits in 3D creating a very unnatural perception. To improve the quality of the rendered image there exist several leads. For example, Devernay *et al.* (2011) propose an artifact detection and removal process whereas Smolic *et al.* (2008) propose to detect unreliable image regions along depth discontinuities and to use a specific processing to avoid the artifacts.

Content Aware Warps Content aware warps methods treat the novel view synthesis problem as a mesh deformation problem. This problem has been extensively studied in the field of media retargeting, where one wants to adapt the images or videos for displays of different sizes and aspect ratios (Wang *et al.*, 2008; Shamir and Sorkine, 2009; Guo *et al.*, 2009). The basic idea is to consider the image as a regular grid, and compute the grid transformation preserving some constraints. In Fig. 3.17 a) we illustrate the results of the grid deformation proposed by Lang *et al.* (2010). To compute the warp they propose to use stereoscopic constraints, temporal constraints as well as saliency constraints. Stereoscopic constraints ensure that the disparity of the resulting image matches the expected mapped disparity. Temporal constraints ensure the warp to evolve smoothly over time. Saliency constraints ensure that the warp preserves as much as possible the shape of detected salient regions (Guo *et al.*, 2008). Less salient regions are allowed to be more distorted. Additionally, Yan *et al.* (2013) propose to add new constraints to preserve lines and planes, as they are likely to seem unnatural when distorted by the warp

(see Fig. 3.17 b) and c)). As we saw in Sec. 2.1.1, the perspective depth cue is mainly guided by straight lines. Moreover, they allow the user to manually add constraints on any region of the scene as some important objects of the scene may be undetected by the saliency estimation. Similar approaches (Chang *et al.*, 2011; Lin *et al.*, 2011) also use content aware warps and allow the user to manually add constraints. Masia *et al.* (2013) also propose a similar method to adapt the content to glasses-free automultiscopic screens.

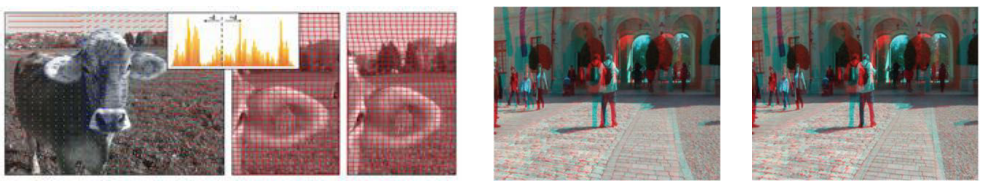
The major advantage of these methods is that they do not create empty disocclusion areas. Every pixel of the target image has a correspondence in the input image, and thus the inpainting hole filling step is avoided. Although some methods (Chang *et al.*, 2011; Lin *et al.*, 2011) claim that the use of sparse features is an advantage with respect to dense disparity maps, the computational cost of GPU stereo methods (Kowalczuk *et al.*, 2013) is nowadays small.

Content aware warps methods have two main limitations. The first is that only moderate modifications of the initial disparity are allowed, i.e. $\times 2, \times 3$ disparity expansion. Otherwise, important stretch artifacts are visible in the final images. The second drawback is that it is unclear how to blend multiple images generated with these techniques. While the blending stage is explicit in DIBR methods, it has never been addressed in the content aware warps literature.

3.4.1 Modifying the Perceived Depth

Let us now study how a disparity mapping function affects the perceived depth from stereopsis. We note the disparity mapping function $\phi(d) : \mathbb{R} \to \mathbb{R}$, transforming a disparity d into a mapped disparity $d' = \phi(d)$. $\phi(d)$ is generally assumed to be increasing monotonic, to avoid mapping farther objects of the scene in front of nearer objects of the scene. As we saw in Sec. 3.1.6, the disparity may be in pixel units or without units, as a fraction of the image size. The function $\phi(d)$ must be, of course, in the proper units. In our work we consider d to be a proportion of the image width, and thus without units.

The mapping function $\phi(d)$ modifies the perceived depth (Eq. 3.33), the ocular divergence limits (Eq. 3.51), the roundness (Eq. 3.62) and the relative perceived size of objects (Eq. 3.71). In the next section we adapt the previous equations by



a) b) c)

Fig. 3.17: *a) Figure reproduced from Lang* et al. *(2010). Stereo correspondences, disparity histogram and close-ups of the warped stereo pair. b) Figure reproduced from Yan* et al. *(2013). With Lang* et al. *(2010) method straight lines are no longer straight. c) Yan* et al. *(2013) method preserves straight lines.*

taking into account $\phi(d)$. With these equations, we can derive constraints on $\phi(d)$.

3.4.1.1 Mapped Perceived Depth from Stereopsis

Let us recall Eq. 3.31 relating the depth z of a scene object to the captured disparity by the filming system with parameters b, H, and W:

$$d = \frac{b}{W}\frac{(z-H)}{z}. \tag{3.74}$$

This disparity value is now mapped with $\phi(d)$ into a new disparity d'. Using Eq. 3.32, which establishes the perceived depth from stereopsis from a disparity, we obtain the mapped perceived depth from stereopsis

$$z' = \frac{H'}{1 - \frac{W'}{b'}\,\phi\left(\frac{b}{W}\frac{(z-H)}{z}\right)}. \tag{3.75}$$

3.4.1.2 Mapped Perceived Position

Given a 3D point in the world $\boldsymbol{x} = (x, y, z)$ and a filming configuration (b, W, H) we project it using $\tilde{\boldsymbol{P}}_f$ (Eq. 3.38) and obtain $\boldsymbol{u} = (u, v, 1, d)$ (Eq. 3.39) that we reproduce:

$$\boldsymbol{u} = \begin{bmatrix} \frac{x}{z}\frac{H}{W} \\ \frac{y}{z}\frac{H}{W} \\ 1 \\ \frac{b}{W}\frac{(z-H)}{z} \end{bmatrix}. \tag{3.76}$$

The disparity component of this vector is now mapped with $\phi(d)$ and we obtain

$$\boldsymbol{u}' = \begin{bmatrix} \frac{x}{z}\frac{H}{W} \\ \frac{y}{z}\frac{H}{W} \\ 1 \\ \phi\left(\frac{b}{W}\frac{(z-H)}{z}\right) \end{bmatrix}. \tag{3.77}$$

With $\tilde{\boldsymbol{P}}_c^{-1}$ (Eq. 3.41) we obtain the mapped perceived 3D point from stereopsis. In homogeneous coordinates it is

$$\tilde{\boldsymbol{x}}' = \begin{bmatrix} \frac{x}{z}\frac{HW'}{WH'} \\ \frac{y}{z}\frac{HW'}{WH'} \\ 1 \\ \frac{b'-\phi(d)W'}{b'H'} \end{bmatrix}. \tag{3.78}$$

The components of the mapped perceived 3D point $\boldsymbol{x'}$ as a function of (b, W, H), (b', W', H'), $\phi(d)$ and the 3D scene point $\boldsymbol{x} = (x, y, z)$ are

$$x' = \frac{x\ b'HW'}{z\ W\left(b' - W'\phi\left(\frac{b}{W}\frac{(z-H)}{z}\right)\right)}, \tag{3.79}$$

$$y' = \frac{y\ b'HW'}{z\ W\left(b' - W'\phi\left(\frac{b}{W}\frac{(z-H)}{z}\right)\right)}, \tag{3.80}$$

and

$$z' = \frac{b'H'}{\left(b' - W'\phi\left(\frac{b}{W}\frac{(z-H)}{z}\right)\right)}. \tag{3.81}$$

3.4.1.3 Mapped Ocular Divergence Limits

In order to avoid ocular divergence, the denominator in Eq. 3.81 should not be negative:

$$\phi(\frac{b}{W}\frac{(z-H)}{z}) \leq \frac{b'}{W'}. \tag{3.82}$$

This condition establishes a maximum value for the mapping function $\phi(d)$. Ocular divergence should be avoided for all elements of the scene. As we assumed $\phi(d)$ to be monotonic, then

$$\phi(d) \leq \frac{b'}{W'} \quad \forall d \in \mathbb{R}. \tag{3.83}$$

3.4.1.4 Mapped Roundness Factor

To compute the mapped roundness factor we first compute the partial derivatives $\frac{\partial x'}{\partial x}$ and $\frac{\partial z'}{\partial z}$. The first one is

$$\frac{\partial x'}{\partial x} = \frac{b'HW'}{z\ W(b' - W'\phi(d(z)))}. \tag{3.84}$$

And the second one is

$$\frac{\partial z'}{\partial z} = \frac{b'H'W'\phi'(d(z))\frac{\partial d(z)}{\partial z}}{(b' - W'\phi(d(z)))^2}, \tag{3.85}$$

where

$$\frac{\partial d(z)}{\partial z} = \frac{b}{W}\frac{H}{z^2}. \tag{3.86}$$

The obtained equation for the mapped roundness factor is

$$\rho(z) = \frac{bH'}{z}\frac{\phi'(d(z))}{(b' - W'\phi(d(z)))}. \tag{3.87}$$

Differentiable Mapping function Let us note that for the mapped roundness factor to be properly defined, we need to impose the differentiable constraint to the disparity mapping function $\phi(d)$. Otherwise, the mapped roundness could not be computed at disparity values where $\phi'(d)$ is not defined. For example, Pitié *et al.* (2012) propose to use disparity mapping functions defined with linear segments. At the junctions points of the linear segments, $\phi'(d)$ is not defined. Of course this can easily be solved creating a smooth transition between both segments. In our work we assume $\phi(d)$ to be differentiable.

3.4.1.5 Mapped Perceived Size

We are now interested in the operator $E_p(z)$ of Eq. 3.68 predicting the puppet-theater effect. We want to see how a disparity mapping function affects its value.

$$E_p(H, z) = \frac{\frac{\partial x'}{\partial x}(H)}{\frac{\partial x'}{\partial x}(z)} \tag{3.88}$$

With Eq. 3.84 we obtain

$$E_p(z) = \frac{z}{H}\frac{(b' - W'\phi(d(z)))}{(b' - W'\phi(d(H)))}. \tag{3.89}$$

3.4.2 Disparity Mapping Functions

Global Linear Mapping The simplest form of disparity mapping function is a linear mapping $\phi(d) = Ad + B$ with $A, B \in \mathbb{R}$, usually written in terms of the maximal and minimal disparity values $(d_{\min}, d_{\max})$, and the maximal and minimal mapped disparity values $(d'_{\min}, d'_{\max})$:

$$\phi_l(d) = \frac{d'_{\max} - d'_{\min}}{d_{\max} - d_{\min}}(d - d_{\min}) + d'_{\min} \tag{3.90}$$

By adapting the interval width of the disparity, the depth range can be scaled and offset to match a target disparity interval. This disparity mapping allows typically to avoid ocular divergence (Eq. 3.83) or to avoid the vergence-accomodation conflict (see Sec. 2.2.1), by choosing

$$\begin{aligned} d'_{\max} &= \frac{b'}{W'}\left(1 - \frac{H'}{z_{\max}(H')}\right), \\ d'_{\min} &= \frac{b'}{W'}\left(1 - \frac{H'}{z_{\min}(H')}\right), \end{aligned} \tag{3.91}$$

where $z_{\max}(H')$ and $z_{\min}(H')$ are given by the empirical results obtained by Shibata *et al.* (2011), Banks *et al.* (2013) and presented in Fig. 2.5.

The image generated with a linear mapping of the disparity corresponds to a new view obtained with *baseline modification*, i.e. the baseline b is scaled with the scalar

A, and the principal point of the camera is shifted with B.

While a global linear mapping allows to constrain the domain of the mapped disparity d', the mapped roundness might be strongly distorted. Let us write Eq. 3.87 substituting $\frac{\partial \phi_l}{\partial d}(d(z)) = A$:

$$\rho(z) = \frac{bH'}{z} \frac{A}{(b' - W'\phi_l(d(z)))}. \tag{3.92}$$

The mapped roundness is scaled accordingly with the factor A.

Global non-linear Mapping Lang *et al.* (2010) propose to use generic nonlinear disparity mapping functions to achieve disparity compression, e.g.

$$\phi_{log}(d) = log(1 + sd) \quad \text{with} \quad s \in \mathbb{R}. \tag{3.93}$$

Devernay and Duchêne (2010) propose a global non-linear disparity mapping function specialized in the adaptation of content from one viewing geometry into another. If the acquisition parameters (b, H, W) are known, the disparity mapping function

$$\phi(d) = \frac{db'H}{bH' + d(HW' - H'W)} \tag{3.94}$$

creates a linear perceived depth transformation with constant roundness factor 1. This can be seen by plugging Eq. 3.94 into Eq. 3.75. We obtain

$$z' = \frac{W'}{W}(z - H) + H'. \tag{3.95}$$

This disparity mapping function corresponds to a viewpoint modification. The transformed images have the same geometry *as if* they were shot with the *Homothetic Setup* (Sec. 3.2.1.2). However, to generate the new views from only two original images is not straightforward. In the first place, one would need to adapt the on-screen size of the scene objects, as a viewpoint modification changes the perspective. Another important problem is that in the viewpoint modification process, large parts of the scene that should be visible may not even be acquired in the original images. Thus large areas of the new views should be inpainted. Devernay and Duchêne (2010) propose an hybrid disparity mapping solution to minimize the inpainted regions.

Locally Adaptive Nonlinear Disparity Mapping As depth in a stereoscopic movie is a narrative tool, it seems appropriate to give the user the control of the depth mapping function. Lang *et al.* (2010) propose to define the disparity mapping function $\phi_a(d)$ as a composition of basic operators ϕ_i, each defined in a different disparity range Ω_i:

$$\phi_a(d) = \begin{cases} \phi_1(d) & \text{if } d \in \Omega_1 \\ \dots & \dots \\ \phi_n(d) & \text{if } d \in \Omega_n \end{cases}. \tag{3.96}$$

Each of these disparity mapping functions can be, either automatically computed by an algorithm, or manually edited by the user. The *Parallax Grading Tool* [1] is a user interaction technique proposed by Pitié *et al.* (2012) allowing the artist to fine tune the final depth of a stereoscopic shot. Chang *et al.* (2011) provide another interactive editing system allowing depth manipulations of the stereoscopic content, e.g. selecting an area and editing its 3D position and scaling factor. All those systems work with locally adaptive nonlinear mapping functions.

3.5 Filming with Long Focal Lengths: Ocular Divergence vs. Roundness

The maximal baseline to avoid ocular divergence is given by Eq. 3.51:

$$b_{\text{div}} = b' \frac{W}{W'}. \tag{3.97}$$

The baseline giving a roundness factor equal to 1 at the depth of the screen $z = H$ is given by Eq. 3.62

$$b_{\text{round}} = b' \frac{H}{H'}. \tag{3.98}$$

We note $f = \frac{H}{W}$ the normalized acquisition focal length, and $f' = \frac{H'}{W'}$ the normalized projection focal length. The ratio between both baselines b_{div} and b_{round} is then equal to the ratio of the normalized focal lengths:

$$\frac{b_{\text{div}}}{b_{\text{round}}} = \frac{f'}{f}. \tag{3.99}$$

As we saw in Sec. 3.2.7, it is reasonable to assume that the normalized projection focal length lies in the interval $[1.4, 2.5]$, 1.4 being the empirical value estimated by Banks *et al.* (2013), and 2.5 being the recommendation in Spottiswoode *et al.* (1952). However, long focal lengths, widely used in live sports broadcast, or nature documentaries, can easily reach normalized focal values around 10, like for example, the "Angenieux Optimo 28-340 cinema lens" (Angenieux, 2015). Acquiring a stereoscopic pair of images with a 340mm focal length does either create ocular divergence, or produce a *cardboard effect* (Sec. 3.2.4.1). Note that this phenomenon is independent of the projection geometry, as the preferred viewing distance depends on the width of the screen. Most stereographers follow the acquisition rules defined by Chen (2012), stating that it is preferable to create a *cardboard effect*, leading to a poor stereoscopic experience, rather than ocular divergence, which creates visual fatigue.

The incompatibility of the divergence baseline and the roundness baseline strongly limits the use of long focal length in today's stereoscopic filming (Mendiburu, 2009).

[1] *parallax* is also used in the cinematographic industry as another term for *disparity*.

3.5.1 Limitations of the State of the Art

As we saw in Sec. 3.4, the literature has addressed the problem to adapt a stereoscopic image to the viewing conditions. To solve the incompatibility between the divergence baseline and the roundness baseline, one could define a disparity mapping function and use those methods.

Using the non-divergent baseline We could acquire the images with the baseline b_{div} from Eq. 3.97 and then use disparity mapping to increase the roundness factor in the desired areas. Unfortunately, in order to add roundness, the disparity map needs to be very accurate to discriminate the local geometry. As the acquiring baseline is small, the obtained precision of the stereo methods is not accurate enough. This can be seen by writing the derivative of the disparity d in Eq. 3.31 with respect to the depth z i.e. how does a change in z affect the disparity:

$$\frac{\partial d}{\partial z}(z) = \frac{b}{W}\frac{H}{z^2}. \tag{3.100}$$

If we use the acquisition baseline b_{div} from Eq. 3.97 and evaluate the derivative at the depth of the screen $z = H$, we obtain

$$\frac{\partial d}{\partial z}(H) = \frac{b'}{WW'}\frac{1}{f}. \tag{3.101}$$

The higher the value of the normalized focal f, the smaller the disparity variation is.

Let us note that Pitié *et al.* (2012) or Didyk *et al.* (2010) are capable to add roundness to the shots, because they work with very accurate, computer generated disparity maps. If they are available, any disparity mapping method could be used.

In the lack of an accurate disparity map, a possible solution would be to use a 2D to 3D conversion technique, like for example the one proposed by Ward *et al.* (2010). The user can select an object and use a depth template, a predefined 3D shape (a sphere, a face, a car), as depth map for the selected object.

Using the roundness baseline Another option would be to acquire the scene with the baseline b_{round} from Eq. 3.98. In this case the problem would be to avoid the divergence created by the farthest elements of the scene. To preserve the acquired roundness, the disparity mapping function would be the identity around the depth of the screen. To avoid ocular divergence, disparities in the background would be compressed, e.g with a nonlinear disparity operator (Lang *et al.*, 2010). In this case the problem is the *visibility*. The disparity of an object of the scene at depth z can be written by combining Eq. 3.31 and Eq. 3.98:

$$d = \frac{b'}{W}\frac{H}{H'}\frac{(z-H)}{z}. \tag{3.102}$$

This disparity value might be very high under certain circumstances. Let us illustrate with a numerical example. If the acquisition parameters are $b_{\text{round}}, H = 20\text{m}$ and $W = 2\text{m}$, and the target projection configuration is a home cinema: $b' = 65\text{mm}$, $H = 3\text{m}$ and $W = 2\text{m}$, then the acquisition baseline is $b_{\text{round}} \approx 48\text{cm}$, and the resulting disparity $d \approx 0.25$, i.e. a 25% of the image size. These high disparity values introduce two important disocclusion areas. The first is near the image borders and the second around depth discontinuities between foreground and background objects. As we illustrate in Fig. 3.18, elements near the right border of the left image are not visible in the right image, whereas elements near the left border of the right image are not visible in the left image. Moreover, background areas near the foreground subject are only visible in one image. The computation of a disparity map from these images can only recover a few disparity values.

3.5.2 Why Do Artists Use Long Focal Lengths?

At this point we have seen that it is not straightforward to generate stereoscopic images with long focal lengths. The natural question for the artists arises: why do artists use long focal lengths? In 2D cinema or television, long focal lengths are used in two cases. The first is when it is impossible to place the camera at the desired position. The second is to create aesthetic perspective deformations of the acquired scene.

The desired camera position is impossible to reach The impossibility to place a camera at the desired position might be physical, or social. For example, when acquiring a live show, the director would like to film the solo of the guitarist of the band with a close shot. However, it might be not acceptable to have cameras on the scene, specially between the performers and the audience. Another example arises when filming a polar bear in the north pole. Although the director would like to have a nice shot of the bear hunting a prey, it would not be safe for the crew (and the equipment) to stand close to the hungry wild beast. In these cases, long focal length allow to create shots, as if we were *close* to the acquired scene, while standing physically far away.

The first motivation of the director to use long focal length is to *get close* to the scene.

Perspective deformations In the cinematography it is well known that different focal lengths distort the perspective, and directors take advantage of these distortions to convey emotions. In Fig. 3.19 we show examples of image distortions when shooting with different focal length. Villains are usually shot with long focal lengths as they appear to be flatter, whereas heroes are shot with a medium focal length to appear rounder. One of the most famous use of the perspective deformation in 2D is the *vertigo effect*, *Hitchcock Zoom* or *dolly zoom*, created by Alfred Hitchcock in 1958 in his feature film *Vertigo*. He compensated the backwards movement of the camera by zooming in the image, to keep constant the size of a target object. Objects in front and behind the target object are strongly distorted

a) b)

Fig. 3.18: *Scene "The Jumper" acquired using two cameras a) and b). The baseline is chosen to create a roundness factor of 1 on the subject (Eq. 3.98). Note how a wide part of the background of image a) is not present in image b), either because it is out of frame, either because it is occluded by the jumper. Inversely, a wide part of the background of image b) is not present in image a). The computation of a disparity map between images a) and b) can only recover few of the background depths.*

while the target object appears to be static. The resulting sequence perfectly conveys the terror of heights felt by the hero. As claimed by 3D professionals (Mendiburu, 2009, 2011), stereoscopy is a narrative tool. Directors should be given the opportunity to play with the perspective distortions at will to create new narratives yet to be invented.

The second motivation of the director to use long focal length is to *add perspective deformations* of the scene.

3.5.3 Proposed Solutions

In this manuscript we propose two different solutions to create stereoscopic shots with long focal lengths, each one following the intentions of the director.

If the director wants to create a shot to *get closer* to the scene, we propose to generate new views with a viewpoint modification (see Sec. 5.1). We propose to acquire the scene with different cameras with different focal lengths and combine them into the desired images. These methods are known as *novel view synthesis* or *free viewpoint rendering* and we address them in Chapter 4.

If the director wants to create a shot to *add perspective deformations* to the scene, we propose to acquire the scene with different cameras, each acquiring the scene with a different baseline, and then combine the images into the final shot (see Sec. 5.2). This idea is not new and is known with the term *multi-rig* (or *multi-rigging*) (Mendiburu, 2009; Devernay and Beardsley, 2010; Dsouza, 2012). The space is divided into n depth regions: $[0, z_1), \ldots, (z_{n-1}, \infty]$. For each region, a baseline b_i is chosen in order to obtain a different perceived depth function $z'(z, b_i)$ (Eq. 3.33). Depending on the depth z of the scene element, the corresponding

Fig. 3.19: *Figure reproduced from Banks et al. (2014). Depth compression and expansion with different focal lengths. A) Left panel: wide-angle effect (short focal length). Picture taken with a 16mm lens (all focal lengths are reported as 35mm equivalent). The goat looks stretched in depth. Right panel: telephoto effect (long focal length). Picture taken with a 486mm focal length. The distance between the pitcher's mound and home plate on an official Major League Baseball field is 18.4 meters. This distance appears compressed. B) Photographs of the same person were taken with focal lengths from left to right of 16, 22, 45, and 216mm. Lens distortion was removed in Adobe PhotoShop, so the pictures are nearly correct perspective projections. Camera distance was proportional to focal length, so the subject's interocular distance in the picture was constant. The subject's face appears rounder with a short focal length and flatter with a long focal length.*

function is used. The final perceived depth function is then

$$z'(z) = \begin{cases} z'(z, b_1) & \text{if } 0 < z \leq z_1 \\ \dots & \dots \\ z'(z, b_n) & \text{if } z_{n-1} < z \leq \infty \end{cases} . \tag{3.103}$$

For example, one could use three cameras as follows. The first two cameras would be placed with a baseline to avoid ocular divergence (Eq. 3.97). Then the third camera would be placed with a baseline with respect to the first to create the desired roundness factor on the subject (Eq. 3.98). In the final shot we would like to have the non-diverging background from the second camera, and the subject with the desired roundness from the third camera. We illustrate the 3 camera multi-rig idea in Fig. 3.20.

Multi-rigging is already used in computer graphics films. Care should be taken in the depth composition of the different layers, specially at the depth transitions z_i between the different shots, as important visible artifacts could appear (Pinskiy *et al.*, 2013). To avoid these artifacts (Pinskiy *et al.*, 2013) propose to use non-linear

a) b) c)

Fig. 3.20: *Scene "The Jumper" acquired using three cameras a), b) and c). The baseline between a) and b) avoids ocular divergence (Eq. 3.97). The baseline between a) and c) creates a roundness factor of 1 on the subject (Eq. 3.98). a) is chosen as the left view of the final stereoscopic pair of images. The right image should ideally be the combination of the subject acquired in image c) (desired roundness factor), and the background acquired in image b) (avoiding ocular divergence).*

viewing rays to ensure smooth transitions between parts of the scene captured with different baselines. If rendering time is not an issue, Kim *et al.* (2011) propose to render a dense lightfield of the scene. Artists have then a per pixel control over the disparity and stereoscopic images can be computed as piece-wise continuous cuts through the lightfield.

Multi-rigging has also been used in actual live steresocopic 3D films, but requires careful planning and important human efforts, as green screens are used to help with the depth composition of the different shots (Dsouza, 2012). Moreover, when planning a multi-rig shot, an "empty safe area" with no scene objects around the compositing depths z_i is used to avoid the visual artifacts(Pinskiy *et al.*, 2013).

In our work we are interested in how to smoothly combine the different shots with different baselines. The depth composition of multiple baseline shots can be interpreted as a disparity mapping function composed from basic operators (see Sec. 3.4.2), with each baseline defining a different disparity mapping function. Moreover, the disparity mapping function could be interpreted in terms of depth. Originally the disparity mapping function was defined in terms of disparity because the main applications of the original approaches, were post-production (Lang *et al.*, 2010) and the content adaptation to the viewing conditions (Devernay and Duchêne, 2010). In both cases the initial input is a stereoscopic image with a range of disparities. However, if we are at the acquisition stage, it is possible not to consider the initial disparity d, and the mapped disparity $d' = \phi(d)$, but the original geometry $z(d)$, and the mapped geometry $z(\phi(d))$. This way we could transform the disparity mapping problem into a more general image-based rendering problem. Of course some adaptations will be needed, as in this mapped geometry the optical rays are not straight anymore. We discuss the proposed solutions in Sec. 5.2.

3.5.4 Research Questions

Although the resulting images from both approaches will be different, both cases share a common problem: how to blend multiple images. The combination of multiple views has been extensively studied in the domain of image-based rendering. In the next chapter we analyze the state of the art and contribute to this domain.

4

Bayesian Modeling of Image-Based Rendering

In the previous chapter we saw that to generate stereoscopic images with a long focal length we need to render novel views of a scene from a given set of input images. In computer graphics this domain is known as Image-Based Rendering (IBR).

In the first section of this chapter we motivate our work and review the state of the art of IBR methods. We also briefly review the state of the art of 3D reconstruction methods, as IBR methods often rely on a geometric knowledge of the scene. We highlight the existing ideas which we build on and describe the current limitations.

In the second section of this chapter we propose a new IBR approach, based on the Bayesian formalism. We detail our approach and conduct experiments to illustrate its benefits and limitations. We also point directions of future improvement.

In the third part of the chapter we establish the formal link between the heuristics widely used in the IBR literature and our model. We conclude the chapter with a summary of the contributions.

4.1 Motivation

In our work, we address the problem of novel view synthesis in the domain of Image-Based Rendering (Shum *et al.*, 2007), where the aim is to synthesize views from different viewpoints using a set of input views in arbitrary configuration. Most of the methods from the state of the art use heuristics to define energies or target functions to minimize, achieving excellent results. A major breakthrough in IBR was the inspiring work of Buehler *et al.* (2001). They define the seven "*desirable properties*" which any IBR algorithm should have: *use of geometric proxies*, *unstructured input*, *epipole consistency*, *minimal angular deviation*, *continuity*, *resolution sensitivity*, *equivalent ray consistency*, and *real-time*. As we will see, those directives still prevail throughout the current state of the art.

Recently, the use of the Bayesian formalism has been introduced in IBR techniques, with the work proposed by Wanner and Goldluecke (2012). They provide the first Bayesian framework for novel view synthesis, describing the image formation process with a physics-based generative model and deriving its Maximum a Posteriori (MAP) estimate. Moreover, their variational method does not only address the problem of novel view synthesis. It directly addresses the synthesis of new super-resolved images, and provides a solid framework for other related problems, namely image denoising, image labeling and image deblurring.

Interestingly, although Buehler *et al.* (2001) and Wanner and Goldluecke (2012) have addressed the same problem, their theoretical results do not converge into a unified framework. On the one hand, the guidelines dictated by Buehler *et al.* (2001) have proven to be very effective, but lack a formal reasoning supporting them. Moreover, it is unclear how the balance between some of the desirable properties should be handled. An illustrative example is the tradeoff between *epipole consistency* and *resolution sensitivity*. The former notes that "*when a desired ray passes through the center of projection of a source camera it can be trivially reconstructed*", while the latter observes that "*in reality, image pixels are not really measures of a single ray, but instead an integral over a set of rays subtending a small solid angle. This angular extent should ideally be accounted for by the rendering algorithm.*" The *epipole consistency* is enforced with an angular deviation term, while the *resolution sensitivity* is driven by the Jacobian of the planar homography relating the views. Both heuristics seem reasonable, but which one should dominate? The choice of the weights between the properties is user-tuned, and in their experiments, parameters have to be adjusted differently depending on the scene.

On the other hand, the existing Bayesian model Wanner and Goldluecke (2012) is able to explain some of the heuristics, but still violates others which seem evident and have proven to work effectively. For example, we do find an analytic deduction of the influence of the foreshortening effects due to the scene geometry in the energy. The findings confirm the heuristic proposed by Buehler *et al.* (2001): it is driven by the Jacobian of the transformation relating the views. However, when carefully analyzing the final equations in Wanner and Goldluecke (2012), an important desirable property proposed in Buehler *et al.* (2001) is still missing: the *minimal angular deviation* of the viewing rays is not enforced and even violated in some cases.

In Fig. 4.1, we illustrate this limitation. In the left part of the figure, we want to render image D with C_1 and C_2. Because of the foreshortening effects, camera C_2 is favored over camera C_1. However, the angular distance of the viewing rays between D and C_1 is much smaller than D and C_2. This is still made more evident in the extreme case where the observed geometry, the camera sensors, and the camera translations are all parallel, as we illustrate in Fig. 4.1b. In this configuration, the contribution of each view is equal, independently of their relative position. However, Buehler *et al.* (2001) desires that the nearest views to the target view, should contribute more than farther views, due to the angular deviation between them.

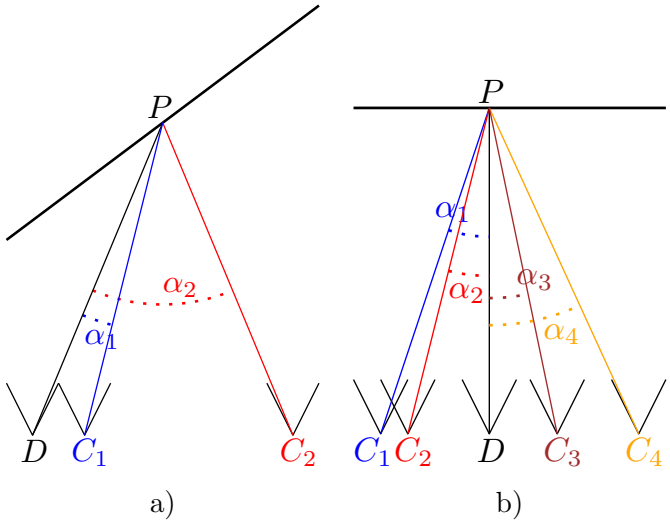


Fig. 4.1: *View D is generated from cameras C_i using Wanner and Goldluecke (2012). a) camera C_2 will be favored over camera C_1 because of the foreshortening effect. However, the angular distance of the viewing rays between D and C_1 is much smaller than D and C_2. b) configuration with a flat scene. All cameras will have the same contribution, despite the different viewing angles.*

Our work is motivated by the differences between state of the art generative models and the energies proposed by generally accepted heuristics. Our goal is to retain the advantage of the intrinsically parameter-free energies arising from the Bayesian formalism, while pushing the image formation model boundaries of Wanner and Goldluecke (2012) and provide a new model which is capable to explain most of the currently accepted intuitions of the state of the art in IBR.

The key point of our method is to systematically consider the error induced by the uncertainty in the geometric proxy. The use of the geometric uncertainty has been inspired by the first desirable property: *the use of a geometric proxy.* According to Buehler *et al.* (2001), an ideal IBR method should be capable to take advantage if geometric information is available and improve the results if the provided geometry is more accurate. For example, in recent years we have seen the arrival of relatively affordable depth sensors, e.g. structured light cameras or time of flight sensors. If those devices provided a better geometry, IBR methods should be capable to integrate their information and improve the rendering results. However, in some specific applications, the computation (or acquisition) of the geometry may not be accurate. To our understanding, the ideal IBR method should also be capable to adapt if only a poor geometric proxy is available. The pursue of this plasticity has led us to consider the geometric uncertainty of the given geometric proxy as an input of our method.

4.2 Related Work

4.2.1 Image-Based Rendering

In 1995, McMillan and Bishop (1995) proposed to consider the different existing image-based rendering techniques as a common problem: the plenoptic sampling. They claimed that movie-maps (Lippman, 1980), image-morphing (Beier and Neely, 1992), view interpolation (Chen and Williams, 1993) and the method proposed by Laveau and Faugeras (1994), could be seen as the attempt to reconstruct the plenoptic function (Adelson and Bergen, 1991) from a sample set of that function. Although IBR methods globally address the same problem, the final purpose of each method, together with the nature of the considered input, still segments most of the existing approaches into image morphing or image view interpolation and free-viewpoint rendering. The taxonomy proposed by Shum *et al.* (2007) shows that most IBR methods rely on an estimation of the geometry, often referred to as "geometric proxy". They propose to classify the methods in an "IBR Continuum" depending on how much geometry they use. In Fig. 4.2 we show the "IBR Continuum", which is well suited to illustrate the variety of IBR methods. On one end of this continuum we have methods which do not use any geometry but rely on a large collection of input images, like light field rendering (Levoy and Hanrahan, 1996), its unstructured version (Davis *et al.*, 2012), and concentric mosaics (Shum and He, 1999). On the opposite end, we have rendering techniques relying on explicit geometry, using accurate geometric models but few images, such as layered depth images (Shade *et al.*, 1998; Chang *et al.*, 1999) and view-dependent texture mapping (Debevec *et al.*, 1998). In between, we find methods using an implicit representation of the geometry, such as view interpolation techniques (Chen and Williams, 1993; Vedula *et al.*, 2005) relying generally on optical flow or disparity maps, transfer methods (Laveau and Faugeras, 1994) establishing correspondences along the viewing rays using epipolar geometry, and the Lumigraph (Gortler *et al.*, 1996), which uses an approximate explicit geometry and a relatively dense set of images.

Naturally, novel view synthesis is prone to produce visual artifacts in regions with a poor (implicit or explicit) reconstruction. Even if the performance achieved

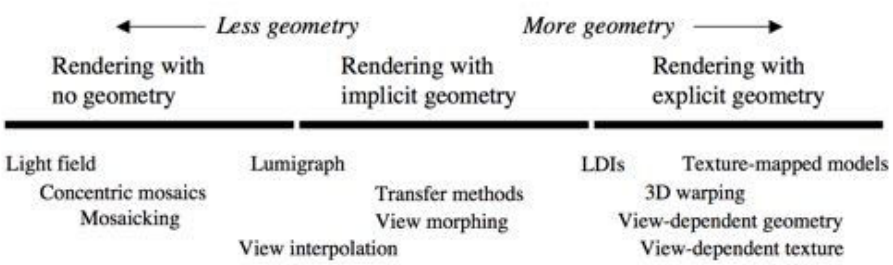


Fig. 4.2: *The IBR Continuum proposed by Shum* et al. *(2007). Methods which do not use any geometry at all (Levoy and Hanrahan, 1996) are on the left end, whereas methods relying on a very precise geometric proxy (Debevec* et al.*, 1998) are shown on the right.*

by state of the art 3D reconstruction methods in estimating geometric proxies is phenomenal, considering them as perfect seems too strong of an assumption: even the best ones have an uncertainty in their final estimates. There are several ways to address the problem, which mainly depend on the target application.

4.2.1.1 Image Morphing Transitions

Image interpolation or image morphing techniques aim at creating compelling transitions between pairs of images. They often rely on an implicit geometric proxy, i.e. optical flow (Chen and Williams, 1993; Wolberg, 1998), although recent evolutions propose extensions of the optical flow (Mahajan *et al.*, 2009) or perceptually based image warps (Stich *et al.*, 2011), both obtaining very impressive results.

In Photo Tourism (Snavely *et al.*, 2006), an interactive tool allowing to browse a large photo collection, non photo realistic view transitions are computed using planes as the geometric proxy. The main difficulty addressed by this work is the difference in appearance between the images, as they may be taken with different cameras and at very different times. With the proposed technique, parallax artifacts arise when the user moves between views. In their followup work (Snavely *et al.*, 2008), the artifacts are reduced by aligning the transition planes with detected features. Taneja *et al.* (2011) propose transitions between cameras recording a dynamic scene. In their work they use billboards as a geometric proxy for the subject of interest, and only use one input view in the rendering. With a clever scheme, they choose when to switch the view to minimize visual artifacts in the transition.

4.2.1.2 View Interpolation

View interpolation aims at generating a new intermediate view between two existing views. Usually these methods use a pair of rectified image and disparity maps as input. We reviewed these methods in Sec. 3.4 and do not discuss them further.

4.2.1.3 Depth Uncertainty Awareness

The idea to use the depth uncertainty in the rendering process is not new. Ng *et al.* (2002) propose a "range-space" approach to compute the possible depths of a pixel and extract the estimated depth uncertainty. The final blend is computed by taking into account the *minimal angular deviation* and the depth uncertainty. With respect to them we seek the inclusion of the *resolution sensitivity* in the blending factors, as well as a formal deduction of the blending weights equations. Hofsetz *et al.* (2004) extend the "range-space" search and propose to extract the depth uncertainty in the form of an ellipsoid. Each ellipsoid is assigned with the color of the input image and the final image is computed by accumulating the projected ellipsoids. They use the blending weights proposed by Buehler *et al.* (2001).

Smolic *et al.* (2008) address the problem of novel view interpolation for multiscopic 3D displays. Although they do not explicitly compute the depth uncertainty, they propose to segment unreliable image regions along depth discontinuities. Unreliable image regions which are prone to introduce visual artifacts are specifically processed.

With the same goal to reduce the visual artifacts arising from a poor geometric reconstruction, Fitzgibbon *et al.* (2005) propose to restrain the space of possible colors with the help of an implicit geometric proxy. For each pixel of the final image, they extract a set of possible color candidates. As the obtained color set includes strong high frequencies between neighboring pixels, they propose to use an image-based prior to select the final best color. During the extraction of the set of possible color candidates, the color contributed by each image is considered independently from the *minimal angular deviation* and the *resolution sensitivity*. Because of the high density of input images, only small artifacts are perceptible in their results.

Goesele *et al.* (2010), in *Ambient Point Clouds*, propose the computation of an improved geometric reconstruction, allowing to detect image regions with poor, or incomplete geometry. For those image regions they propose to use a non-photo-realistic transition based on epipolar constraints.

4.2.1.4 Dense Camera Arrays

Another way to address this problem is to improve the acquisition setting, and use a relative high density of images, as done by Zitnick *et al.* (2004) and Lipski *et al.* (2010). They achieve a good enough reconstruction, leading to impressive novel view synthesis. However, their setting is heavily constrained.

Lipski *et al.* (2014) propose an hybrid approach between image-morphing and depth-image-based rendering, including a refinement of the explicit geometry and the implicit correspondence estimation. They considerably improve the blur artifacts created by small inaccurate registrations of the warped images, but their method strongly relies on precise image correspondences, which are not available with wide-baseline configurations.

Although free-viewpoint navigation is possible with those techniques, the novel view locations are often constrained to positions between the input views and do not allow to the virtual camera to "get closer" to the scene.

4.2.1.5 Free Viewpoint Rendering

The approaches of Kanade *et al.* (1997) and Moezzi *et al.* (1997) are known to be the earliest 3D video multi camera studios for free-viewpoint rendering. The main idea is that during rendering, the multiple images can be projected onto a geometric proxy, in order to generate a realistic view-dependent surface appearance (Matusik and Pfister, 2004; Carranza *et al.*, 2003; Tanimoto, 2012). The ability to interactively control the viewpoint during rendering has been termed *free-viewpoint video* by the MPEG Ad-Hoc Group on 3D Audio and Video (Smolic and McCutchen,

2004; Smolic *et al.*, 2005). A *free-viewpoint rendering* method should be capable of handling wide-baseline camera configurations and not constraining the position of the novel rendered views (Zinger *et al.*, 2010), as is usually the case in image interpolation methods.

Most of the contributions in this domain have targeted the productions of live events, and in particular sports. Because of the heavy constraints in the live broadcast settings, most methods address all the difficult problems of camera calibration, reconstruction and rendering in an unified framework. For example, Germann *et al.* (2012) present a complete solution to the novel view synthesis problem, performing acquisition, reconstruction and rendering.

An interesting approach related to our purpose is the work from Hilton *et al.* (2011), where they propose to render stereoscopic images from a standard camera configuration used for a 2D broadcast. In a standard 2D broadcast camera setup the number of cameras can be relatively high (up to 26), thus they propose to combine them and generate stereoscopic shots. In their work they do not explicitly address the long focal length shots, and our approach to the stereoscopic zoom (Chapter 5), could be build on such a framework.

Most interestingly, there has been an evolution of the geometric proxys used in the literature of free-viewpoint rendering in the sports domain: Hayashi and Saito (2006) propose to use billboards, which result in blurry images, or ghosts, because of small errors in the image registration. Grau *et al.* (2007) propose to use the visual hull from silhouettes, which has the limitation that a small error in the camera calibration can remove thin structures like arms and legs. Germann *et al.* (2010) extend the billboards to articulated billboards, which usually rely on interactive pose estimation algorithms. According to Guillemaut *et al.* (2009) and Hilton *et al.* (2011), those algorithms ask for too much user interaction to be practical for long sequences. Therefore Guillemaut *et al.* (2009) and Guillemaut and Hilton (2011) propose an approach to jointly optimize scene segmentation and player reconstruction from silhouettes, taking into account camera calibration errors. When observing the past evolution of the geometric proxys, and expecting new evolutions to appear, we believe that an ideal IBR method should be capable to adapt to and benefit from the improved geometric proxys.

4.2.1.6 Unstructured IBR

In the literature addressing the generic IBR unstructured configurations outside the sports domain, Hornung and Kobbelt (2009) propose to improve the rendering quality by computing the warps from the input views onto the target views using a particles approach. Their improved reconstruction allows them to create new views from different view positions and different focal distances. In the blending stage they use the weights proposed by Buehler *et al.* (2001).

Although most methods use either an implicit or an explicit geometric proxy, some approaches propose an hybrid approach considering both kinds of geometries. For example, Floating Textures (Eisemann *et al.*, 2008) propose a first match between textures using an explicit geometry, which is then adjusted using the optical flow

between the input images. In the proposed framework, any weight can be considered in the blending stage, as long as they are normalized: the sum of all contributing camera must be 1.

Chaurasia *et al.* (2013) deal with inaccurate or missing depth information proposing local shape-preserving warps based on superpixels. An over-segmentation of images allows them to create plausible renderings for scene regions with unreliable geometry. In the blending stage they use the weights from Buehler *et al.* (2001) with a supplementary modification: to avoid excessive blending they only blend 2 views. Kopf *et al.* (2014) propose to create first person hyperlapse videos from a video sequence. They reconstruct a geometric proxy and compute a new trajectory for the camera taking into account the guidelines of Buehler *et al.* (2001). The final fusion of the images is performed as a labeling problem (Agarwala *et al.*, 2004). They contribute an improvement on how to enforce the resolution penalty. Instead of computing the determinant of the Jacobian of the warp, which can be small even for highly distorted views, they propose to individually use the singular values of the Jacobian matrix, which better account for the stretch of the image. In their work they do not account for the *minimal angular deviation*, as the movement in their input images is mostly frontal. In a sequence with a lateral movement, a flat geometry would have the same penalty for all views as we illustrated in Fig. 4.1b.

4.2.1.7 How to Blend Multiple Images?

When Buehler *et al.* (2001) introduced Unstructured Lumigraph Rendering, they established the seven "*desirable properties*" that all IBR methods should fulfill: *use of geometric proxies*, *unstructured input*, *epipole consistency*, *minimal angular deviation*, *continuity*, *resolution sensitivity*, *equivalent ray consistency*, and *real-time*. In their work they reviewed the best eight performing methods at the time (Levoy and Hanrahan, 1996; Gortler *et al.*, 1996; Debevec *et al.*, 1996; Pighin *et al.*, 1998; Pulli *et al.*, 1997; Debevec *et al.*, 1998; Heigl *et al.*, 1999; Wood *et al.*, 2000) and observed that none of them fulfilled all the "*desirable properties*". For example, none of them considered the *resolution sensitivity* property: *"In reality, image pixels are not really measures of a single ray, but instead an integral over a set of rays subtending a small solid angle. This angular extent should ideally be accounted for by the rendering algorithm"* (Buehler *et al.*, 2001). Only half of the studied methods take into account the *minimal angular deviation*: *"In general, the choice of which input images are used to reconstruct a desired ray should be based on a natural and consistent measure of closeness. In particular, source images rays with similar angles to the desired ray should be used when possible"* (Buehler *et al.*, 2001). Consequently they proposed a new method with heuristics enforcing the guidelines. This work has been of major importance in the community. The proposed guidelines have been adopted by most of the IBR methods and still prevail in recent work (Hornung and Kobbelt, 2009; Chaurasia *et al.*, 2013; Kopf *et al.*, 2014).

In our work we focus on the *desirable properties* directing which image should be preferred over the others, also known as the blending weights. Those properties are the *minimal angular deviation*, the *resolution sensitivity* and the *continuity*.

Although Bayesian formalisms are a common way to deal with spatial super-resolution in the multi-view and light field setting (Bishop and Favaro, 2012; Goldluecke and Cremers, 2009), they have only recently been introduced to IBR with the work by Wanner and Goldluecke (2012). While their work provides a physical explanation for the *resolution sensitivity* property, the *minimal angular deviation* can be violated in their final equations. Most interestingly, Vangorp *et al.* (2011) empirically verify which properties in IBR methods are prone to create visual artifacts, and one of their main results identifies the *minimal angular deviation* as a key property to be taken into account to avoid visual artifacts.

Takahashi (2010) studies the theoretical impact of errors in the geometric proxy when rendering a new view from 2 images. In their results they obtain the *minimal angular deviation* as the optimal blend between two images. However, as their approach is restrained to only 2 views, there is no insight on how the camera resolution should be taken into account.

Raskar and Low (2002) propose a finer description of the *continuity* desirable property, by establishing guidelines to achieve spatial and temporal smoothness: *normalization* (sum of weights should equal 1), *scene smoothness*, *intra-image smoothness*, *near view fidelity* (grouping the *epipole consistency* and *minimal angular deviation* from Buehler *et al.* (2001)) and *localization*. In order to achieve the desired continuity, they propose to consider two different contributions for each view: one is view-dependent, and the other is view-independent. The view-dependent contribution is enforced using a similar penalty as Buehler *et al.* (2001). The view-independent contribution is computed in each view by identifying depth discontinuities in the image using a threshold and computing the distance of the pixels to the detected depth discontinuity. The view-independent heuristic highly improves the results near occlusion boundaries. In our work we were inspired by the view-independent heuristic and aim at avoiding the depth discontinuity threshold, as well as to provide a formalization on why pixels near a depth discontinuity should be penalized.

Similarly, Takahashi and Naemura (2012) propose a view-independent weighting method. They propose to use the confidence on the depth estimates, or as we call it, the depth uncertainty, in their "Depth-Reliability-Based Regularization". Instead of weighting the contributing pixels with different weights, they act on the balance between the regularizer and the data term. The reconstruction of a ray corresponding to an unreliable depth measure is mainly guided by the regularizer term, whereas the reconstruction of a ray corresponding to a reliable depth measure is mainly guided by the data term. Artifacts due to unreliable depth measures are thus avoided. What drives our attention in the proposed work is the use of the computed depth uncertainty, which is less reliable near depth discontinuities. They provide a way to eliminate the threshold to compute the distances of pixels to depth discontinuities used by Raskar and Low (2002). Surprisingly, the weights in their work neither consider the *minimal angular deviation*, the *resolution sensitivity*, nor the *continuity* properties.

To summarize, in the literature addressing the problem of how to blend multiple images, we have Wanner and Goldluecke (2012) who provide formal insights to the

resolution sensitivity, and Takahashi (2010) who provides formal insights to the *minimal angular deviation*. We did not find any formal insights on the *continuity* property.

4.2.2 3D Reconstruction Methods

As we have seen in the previous section, IBR techniques are strongly related to the geometric proxy. Thus we briefly review the 3D reconstruction methods. We focus on the explicit 3D reconstructions, which are the most generic with respect to the camera configuration. For a detailed review of the 3D reconstruction methods we refer the reader to Seitz *et al.* (2006).

4.2.2.1 Explicit 3D Reconstructions

An explicit geometric proxy may represent the 3D shape of a scene in different ways: depth maps, meshes, point clouds, patch clouds, volumetric models and layered models, each representation having its own advantages and drawbacks. Let us briefly introduce them. The *multiple-baseline stereo* problem was addressed by Okutomi and Kanade (1993). In this configuration all cameras are aligned, a reference view is chosen and the distances between each camera and the reference view are called the baselines. One of the advantages is that using multiple images reduces the ambiguity of matching. The drawback is that computations are done with respect of the reference view. We only obtain the geometry *as seen* from this reference view. Moreover, large baselines have problems in the matching steps because of occlusions.

The *volumetric stereo* or *voxel coloring* approach computes a cost function on a 3D volume, and then extracts a surface from this volume. The goal is to assign a color to each of the voxels. Space carving algorithms (Seitz and Dyer, 1999; Kutulakos and Seitz, 2000; Bonfort and Sturm, 2003; Furukawa and Ponce, 2006) are a popular approach to this problem. Their main advantage is that their result is a reasonable initial mesh that can then be iteratively refined. Most of the approaches rely on a silhouette extraction stage, which makes it difficult for them to precisely extract the rims of the scene. Moreover, as an important part of the scene is empty, a lot of computations are performed on voxels that are not on the scene.

Another way to create a 3D reconstruction is to rely on a set of sparse features matched in the input images. Those features are merged into tracks corresponding to 3D points of the scene. For example, Shahrokni *et al.* (2008) propose to create a coarse 3D model of the scene by creating solid triangles with the 3D points as vertices. Similarly, Furukawa and Ponce (2010) propose to first extract features and get a sparse set of initial matches. Those matches are iteratively expanded to nearby locations, and false matches are filtered out using visibility constraints. A great advantage of this method is that it can scale with an increasing number of images (Furukawa *et al.*, 2010), thus being capable to reconstruct large scenes with a high number of input cameras. Moreover, the expanded matches can be merged

into a 3D surface with surface reconstruction techniques, like for example *Poisson reconstruction* (Kazhdan *et al.*, 2006).

Another alternative is to create simpler 3D models based on piece-wise planar proxies, which have been demonstrated in the modeling of interior scenes (Furukawa *et al.*, 2009) as well as exterior scenes (Sinha *et al.*, 2009).

Depending on the application, ad-hoc acquisition devices can be used, either with a specific custom device (Kim and Hilton, 2009) or directly with depth scanners, e.g. structured light sensors or time of flight cameras.

4.2.2.2 Depth Uncertainty

In our work, in addition to the 3D reconstruction, we would like to have access to the geometric uncertainty of the 3D reconstruction. By geometric uncertainty, we mean a geometric measure in world units describing the possible deviation in the measured position of the 3D point.

The uncertainty associated with depth measures has been studied in the field of robotics, e.g. to address the problem of depth fusion of new measurements with old ones (Matthies *et al.*, 1989), as well as in the literature of stereo disparity computation (Kanade and Okutomi, 1994; Fusiello *et al.*, 1997). In general, reconstruction methods provide a confidence measure in the form of a score, often associated to the photo-consistency of the 3D point when projected into the images. This score should be used with caution, because the confidence measure from a depth estimator is usually unit-less, whereas the geo-uncertainty is in world units. Although tempting, one should avoid to take the score measures *as* the geometric uncertainty, as stated by the study performed by Hu and Mordohai (2012).

However, some algorithms already provide the uncertainty information of the geometric reconstruction. For example, reconstruction methods using probabilistic inference (Gargallo *et al.*, 2007; Liu and Cooper, 2014), compute the entire probability distribution of the 3D reconstruction. By analyzing the probability distribution one can deduce the geometric uncertainty of the estimated depth, which is usually the 3D reconstruction corresponding to the MAP configuration. As we saw earlier, Ng *et al.* (2002) and Hofsetz *et al.* (2004) propose a "range-space" method to compute the uncertainty associated to the computed depth of a pixel. The computed volumetric depth uncertainty could be integrated as the input of our method.

Unfortunately, most 3D reconstruction methods do not provide an estimate of the uncertainty of the geometric proxy. In Sec. 4.5.1.3 we propose simple approaches to estimate it.

A comment on the learning approach to estimate the geometric uncertainty. Before we detail our generative model, we would like to point a learning approach proposed by Mac Aodha *et al.* (2010) and Reynolds *et al.* (2011) to estimate the accuracy of any image processing algorithm. Their work hypothesis is that the accuracy of the algorithm depends on the input data. They treat the

algorithm as a black box, feeding it with controlled data and analyzing the output result. Then, comparing the results with the groundtruth, they obtain an error map. Then they train classifiers in order to find patterns between the input data and the error map. Once the classifiers are trained, they can first analyze the input data, and predict the accuracy of the method. Furthermore, Mac Aodha *et al.* (2010) propose to use multiple algorithms on the same input image. They first segment the input image, by assigning the best predicted algorithm to each part of the image. Then they only apply the corresponding algorithm to the segmented area. While effective, the main limitation of these methods is the amount of initial data to train the classifiers. However with the impressive growth of the available data, it is indeed a promising lead in the estimation of the accuracy of any image processing algorithm.

4.3 Formalizing Unstructured Lumigraph

In this section we briefly introduce the Bayesian formalism for the IBR problem. Then we describe the proposed novel view synthesis generative model.

4.3.1 The Bayesian Formalism

Probability theory provides an ideal framework to formalize inverse problems. The idea is to jointly model the observed data and the unknown variables in a single probability space. Having such a space one can simply ask the question: what is the probability of a solution given the observed data? Formalizing real problems in this way is often called the Bayesian approach.

To take this approach we were inspired by the work of Mumford (1994), providing a Bayesian rationale for the image segmentation problem, the work of Gargallo I Piracés (2008), providing a Bayesian rationale for the multi-view stereo problem, and the work of Wanner and Goldluecke (2012), providing the first Bayesian rationale for the IBR problem. Let us present the general Bayesian rationale to the IBR problem.

In image-based rendering, the observed variables are the pixel values of the input images v_i, and the geometric proxy of the world g. The unknown variables are the pixel values of the target image u. The joint probability of input images, the geometry and target image $p(v_i, g, u)$ is a distribution on the space of all possible images and all possible geometries. We would like to encode all our knowledge of the problem in this distribution. Given a set of input images, the geometry and a target image, we should be capable to measure how plausible the set is to us. For example, if in the input images there are green trees, the target image should be likely to contain green trees. Because defining a joint distribution is very difficult, approximations are done by decomposing the distribution in simpler terms:

$$p(v_i, g, u) = p(v_i, u|g)p(g), \tag{4.1}$$

and

$$p(v_i, u|g) = p(v_i|u, g)p(u), \tag{4.2}$$

where $p(v_i|u, g)$ is the conditional probability of the input images given the target image and the geometric proxy. This distribution is called the *likelihood* and aims to quantify the question: *if the target image is u and the geometric proxy is g, how likely is to observe v_i?*

The terms $p(g)$ and $p(u)$ are known as the *prior* and should quantify the question: *is the target image (or the geometric proxy) plausible?*

This sort of decomposition is called a *generative model* and is an obvious model to formalize inverse problems. The question we would like to answer is: *given a geometric model and a set of input images, how probable is the target image?* The answer is the posterior distribution $p(u|g, v_i)$. Using the Bayes' theorem, the

posterior distribution can be written with the help of the joint distribution as

$$p(u|g, v_i) = \frac{p(g, v_i, u)}{p(g, v_i)} = \frac{p(g, v_i|u)p(u)}{\int p(g, v_i|u)p(u)du} = \frac{p(v_i|g, u)p(g)p(u)}{\int p(g, v_i|u)p(u)du}. \tag{4.3}$$

This relation is very valuable because it relates what we know, the likelihood and the prior, to what we want, the posterior. Usually, at the end, we are not interested in the entire distribution of the target images, but a single final image would be preferred. A common choice is to select the most probable target image, which is called maximum a posteriori (MAP).

The MAP target image is obtained by minimizing the negative logarithm of the probability, which is referred to as the *energy*:

$$E(u) = -\ln p(u|g, v_i) \tag{4.4}$$

$$= -\ln p(v_i|g, u) - \ln p(u) - \ln p(g) \tag{4.5}$$

$$= E_d(v_i, g, u) + E_r(u) + E_r(g). \tag{4.6}$$

The log of the prior term $p(u)$ (or $p(g)$) is often called the *regularizer*, as it was originally conceived to make the minimization of E_d well-posed. The log of the likelihood term is often called the *data term*, as it dependents on the observed data. In the IBR Bayesian rationale we are interested in the target image u, and the geometry is considered to be an input. Hence the probability of g is a constant and $E_r(g)$ does not play a role in the minimization process.

4.3.2 Novel View Synthesis Generative Model

Our goal is to synthesize a (possibly super-resolved) view $u : \Gamma \to \mathbb{R}$ from a novel viewpoint c using a set of images $v_i : \Omega_i \to \mathbb{R}$ captured from general positions c_i. We assume we have an estimate of a geometric proxy which is sufficient to establish correspondence between the views. More formally, the geometric proxy induces a backward warp map $\tau_i : \Omega_i \to \Gamma$ from each input image to the novel view, as well as a binary occlusion mask $m_i : \Omega_i \to \{0, 1\}$, which takes the value one if and only if a point in Ω_i is visible in Γ. If we restrict τ_i to the set of visible points $V_i \subset \Omega_i$, it is injective and its left inverse, the forward warp map $\beta_i : \tau_i(V_i) \to \Omega_i$ is well defined (see Fig. 4.3).

4.3.2.1 Ideal Image Formation Model

In order to consider the loss of resolution from super-resolved novel view to input view, we model the subsampling process by applying a blur kernel b in the image formation process of v_i. We note $\hat{v}_i$ as the continuous collection of rays, and apply the point spread function (PSF) of camera i to obtain the image

$$v_i = b * \hat{v}_i. \tag{4.7}$$

Each pixel of v_i stores the integrated intensities from a collection of rays from the scene, and the novel view u is always considered to have a higher resolution than the input views.

Let us discard the effects of visibility for a moment, supposing all points are visible. Also suppose we have a perfect backward warp map τ_i^* from Ω_i to Γ, and perfect input images v_i^* and $\hat{v}_i^*$. Assuming the Lambertian image formation model, the idealized exact relationship between novel view and input views is

$$v_i^* = b * \hat{v}_i^* = b * (u \circ \tau_i^*), \tag{4.8}$$

being $\circ$ the function composition operator. However, the observed images v_i and geometry τ_i are not perfect, and we need to consider these factors in the image formation model.

4.3.2.2 Sensor Error and Image Error

First, we consider the sensor error ε_s, and we assume it follows a Gaussian distribution on all cameras with zero mean and variance σ_s^2. While the sensor noise variance σ_s^2 and the subsampling kernel b could be different among views, for the sake of simplicity of notation, we assume them to be identical for all cameras.

Second, we consider the error in the geometry estimate, which implies that the corresponding backward warp map τ_i is different from the ideal map τ_i^*. This induces an intensity error ε_{g_i} in the image formation process,

$$\varepsilon_{g_i} = b * (u \circ \tau_i^*) - b * (u \circ \tau_i). \tag{4.9}$$

This error can also be written as

$$\varepsilon_{g_i} = b * \hat{\varepsilon}_{g_i} = b * (\hat{v}_i^* - \hat{v}_i), \tag{4.10}$$

where $\hat{\varepsilon}_{g_i}$ is the super-resolved intensity error.

The uncertainty related to the intensity error ε_{g_i} is denoted by $\sigma_{g_i} : \Omega_i \to \mathbb{R}$. Note that both have intensity units.

Taking into account the above errors, the image formation model becomes:

$$v_i = b * (u \circ \tau_i + \hat{\varepsilon}_{g_i}) + \varepsilon_s. \tag{4.11}$$

While we make the common assumption that ε_s follows a Gaussian distribution, the distribution of $\hat{\varepsilon}_{g_i}$ is yet unknown to us. What we know is that $\hat{\varepsilon}_{g_i}$ is strongly related to the geometric error. In the next section, we study the relationship between their distributions.

4.3.2.3 Dependency of Image Error on Geometric Error

The geometric proxy yields for each point x in Ω_i a depth measure $\hat{z}_i(x)$ and its associated uncertainty $\hat{\sigma}_{z_i}(x)$, giving us a distribution of depth along the viewing

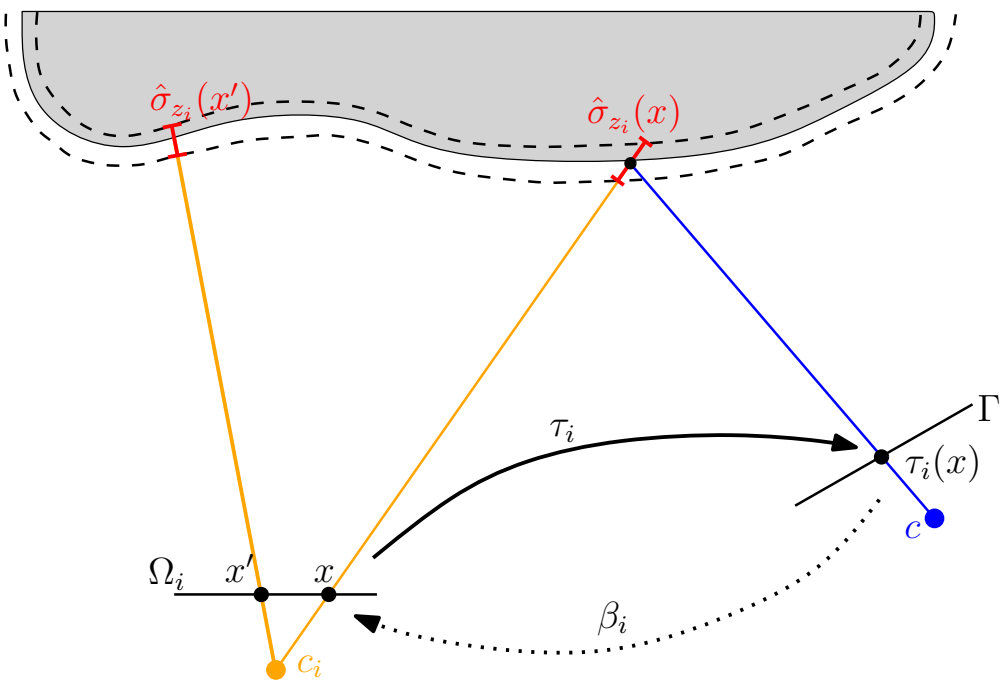


Fig. 4.3: *Transfer map τ_i from image plane Ω_i into target image plane Γ. The depth uncertainty σ_{z_i} may be different among pixels.*

ray from c_i, as illustrated in Fig. 4.3. We write the subsampled uncertainty as

$$\sigma_{z_i} = b * \hat{\sigma}_{z_i}. \tag{4.12}$$

We now consider the error $\hat{\varepsilon}_{z_i}$ in the estimation of the geometric proxy, expressed in world units. The previous image error $\hat{\varepsilon}_{g_i}$ is dependent on the underlying geometric error. Note that the image error has intensity units and must not be confused with $\hat{\varepsilon}_{z_i}$ having geometric units. In contrast to the blur kernel and the sensor noise, we allow these errors to be different for each view and for each pixel in each view, as made explicit in the notation.

We assume that the error distribution for the depth estimates is normal, $\hat{\varepsilon}_{z_i} \sim \mathcal{N}(0, \hat{\sigma}_{z_i}^2)$. The goal is now to derive how this distribution generates a color error distribution in the image formation process. Propagating a distribution with an arbitrary function is not straightforward, even if in our case, this depth error distribution is assumed to be Gaussian, and is only propagated along the epipolar lines.

In the case where the function is monotonic (increasing or decreasing), then the transformation of a probability distribution can be computed in closed form. So, instead of computing the full color distribution along the viewing ray, we linearize and consider the first order Taylor expansion of $\hat{v}_i$ with respect to z_i. This implies that the resulting color distribution is also Gaussian, with mean $\mu_i = u \circ \tau_i$ and

standard deviation

$$\hat{\sigma}_{g_i} = \hat{\sigma}_{z_i} \left| \frac{\partial \hat{v}_i}{\partial z_i} \right| . \tag{4.13}$$

Using Eq. 4.8 and the chain rule, we find that

$$\hat{\sigma}_{g_i} = \hat{\sigma}_{z_i} \left| \frac{\partial (u \circ \tau_i)}{\partial z_i} \right| = \hat{\sigma}_{z_i} \left| (\nabla u \circ \tau_i) \cdot \frac{\partial \tau_i}{\partial z_i} \right| . \tag{4.14}$$

As $\hat{\sigma}_{z_i}$ is always positive, the subsampled color variance σ_{g_i} can be written as

$$\sigma_{g_i} = b * \left| (\nabla u \circ \tau_i) \cdot \hat{\sigma}_{z_i} \frac{\partial \tau_i}{\partial z_i} \right| . \tag{4.15}$$

MAP estimate and energy In the Bayesian formulation, the MAP estimate of the novel view can be found as the image u minimizing the energy

$$E(u) = E_{\mathrm{d}}(u) + \lambda E_{\mathrm{r}}(u), \tag{4.16}$$

where the data term $E_{\mathrm{d}}(u)$ is deduced from the generative model, and $E_{\mathrm{r}}(u)$ is a smoothing term which is detailed afterwards. $\lambda > 0$ is the only parameter of our method, and it controls the smoothness of the solution.

Let us consider the two error sources as independent, additive and Gaussian. Then their sum is also a normal distribution with zero mean and variance $\sigma_s^2 + \sigma_{g_i}^2$. The data term computed from the generative model of Eq. 4.11 is given by:

$$E_{\mathrm{d}}(u) = \sum_{i=1}^{n} \frac{1}{2} \int_{\Omega_i} \omega_i(u)\, m_i (b \ * \ (u \circ \tau_i) - v_i)^2 \, dx, \tag{4.17}$$

$$\text{with } \omega_i(u) \ = \left(\sigma_s^2 + \sigma_{g_i}^2\right)^{-1} . \tag{4.18}$$

This data term is similar to the one found in the previous model from Wanner and Goldluecke (2012), except for the factor $\omega_i(u)$, which can be seen as a weight that depends both on the depth uncertainty and on the latent image u being computed. If there were no depth uncertainty, this term would reduce to σ_s^2, which gives exactly the energy found in Wanner and Goldluecke (2012). Let us remark our abuse of notation when writing $\omega_i(u)$. The function ω_i is defined as $\omega_i : \Omega_i \to \mathbb{R}$. Our purpose with the notation is to make explicit the dependency on the latent image u. In order to optimize the final energy we compute the Euler-Lagrange equations of our functional. The fact that ω_i depends on u is important.

Interesting observations From Eq. 4.15, we can observe that the term $\sigma_{g_i}^2$ in $\omega_i(u)$ becomes smaller if the length of the vector $\partial \tau_i / \partial z_i$ decreases. The derivative $\partial \tau_i / \partial z_i$ denotes *how much the reprojection of a point x_i from the original view v_i onto the novel view u varies when its depth $z_i(x_i)$ changes.* This vector points towards the direction of the epipolar line on u issued from the point x_i of v_i, and its magnitude decreases with the angle between the optical ray issued from

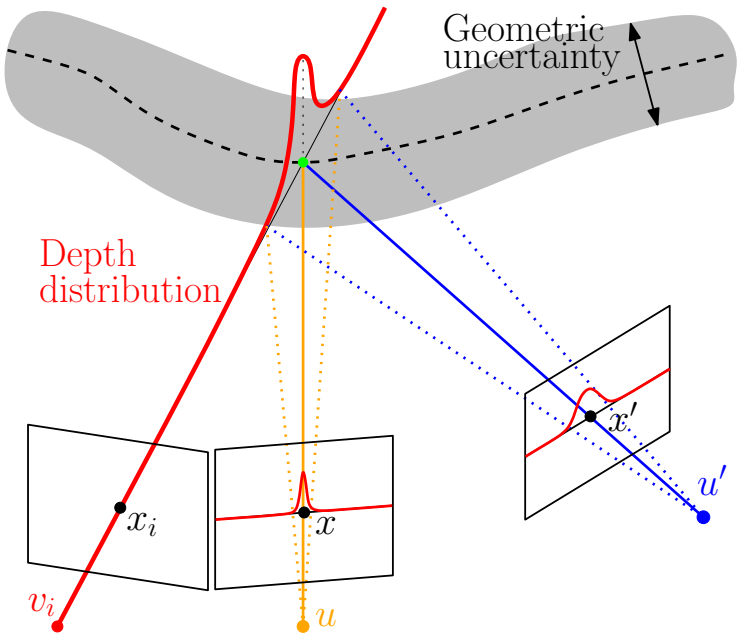


Fig. 4.4: *A depth distribution along an optical ray of camera v_i propagates differently depending on the viewing angle of the rendered camera u or u'. The bigger the angle, the bigger the projected uncertainty will be.*

the original view v_i and the optical ray from the novel view u. As illustrated in Fig. 4.4, the term $\sigma_{g_i}^2$ thus accounts for the *minimal angular deviation* "*desirable property*" from Buehler *et al.* (2001), which was not accounted for in Wanner and Goldluecke (2012).

Let us analyze more precisely under which circumstances the weight $\omega_i(u)$ reaches its maximal value $1/\sigma_s^2$. There are three situations in which this occurs. The first one is if $\partial\tau_i/\partial z_i = 0$, i.e. the depth of a point in v_i has no influence on its reprojection onto u. This can only happen if the two optical rays are identical, which corresponds to the *epipole consistency* property from Buehler *et al.* (2001). The second one is if $\nabla u = 0$, i.e. the rendered image has no gradient or texture at the considered point: in this case, an error on the depth estimate has no effect on the rendered view. The last situation is if ∇u at the rendered point is orthogonal to the direction of the epipolar line from camera i passing through the rendered point: a small error on the depth estimate in camera i does not have an effect on the rendered view because the direction of influence of this error is tangent to an image contour in u.

4.3.2.4 Choosing the Prior

The prior is introduced in the Bayesian formulation (see Sec. 4.3.1) to restrain the possible configurations of the target image. The obtained regularizer allows to overcome the ill-posedness of the minimization problem. For example, in the super-resolution problem, the ill-posedness can be studied by analyzing the dimension

of the null-space of the matrix system. In the analysis performed by Baker and Kanade (2002), the authors show that the dimension of the null-space of the matrix system increases with an increase of the super-resolution factor. Furthermore, in novel view synthesis, some parts of the image may not be seen by any contributing view. The regularizer allows to fill the gaps with plausible information. Thus, the choice of the prior has significant influence on the final result.

Very interesting priors have been developed in order to overcome specific issues in super-resolution. For example Shan *et al.* (2008) propose to impose smoothness on the final image on areas where the input images are also smooth. There are also techniques allowing to learn generic image priors from a collection of images Roth and Black (2005). As we deal with a potentially (very) large set of input images, those techniques could be applied. However, the focus of this work is on the generative model. We use basic total variation as a regularizer,

$$E_{\mathrm{r}}(u) = \int_{\Gamma} |Du| \,, \tag{4.19}$$

which is convex and has been extensively studied in the context of image analysis problems (Chambolle, 2004). The search for optimal priors is left as a topic of future work.

4.3.2.5 Optimization

The energy from Eq. 4.16 has integrals in different domains. The first step is to do a variable substitution of the data term of Eq. 4.17 so that both terms have the same domain. We perform the variable substitution

$$\begin{cases} x = \beta_i(y) \\ dx = |\det D\beta_i| dy, \end{cases} \tag{4.20}$$

where $D\beta_i$ denotes the Jacobian matrix of β_i.

The obtained expression is:

$$E_{\mathrm{d}}(u) = \sum_{i=1}^{n} \frac{1}{2} \int_{\Gamma} |\det D\beta_i| \ \left(\omega_i(u)\ m_i\ (b\ *\ (u \circ \tau_i) - v_i)^2\right) \circ \beta_i\ dy. \tag{4.21}$$

The energy from Eq. 4.16 is hard to optimize because the weights $\omega_i(u)$ in Eq. 4.21 are a nonlinear function of the latent image u. The Euler-Lagrange equations are not straightforward because of this dependence. In order to overcome this limitation we propose a re-weighted iterative method similar to the one proposed by Cho *et al.* (2012). We use an estimate $\tilde{u}$ of u, set at $\tilde{u} = \frac{1}{n}\sum v_i \circ \beta_i$ in the first iteration. Then we consider $\omega_i(\tilde{u})$ constant during each iteration, making the simplified energy convex.

Furthermore, with arguments similar to Wanner and Goldluecke (2012), we can

show that the functional derivative of the simplified data term is

$$dE^i_{\mathrm{d}}(u) = |\det D\beta_i| \left(\omega_i(\tilde{u})\ m_i\ \bar{b} * (b*(u \circ \tau_i) - v_i)\right) \circ \beta_i, \tag{4.22}$$

where $\bar{b}(x) = b(-x)$ is the adjoint kernel. This functional derivative is Lipschitz-continuous, which allows to minimize the energy via the fast iterative shrinkage and thresholding algorithm (FISTA) proposed by Beck and Teboulle (2009). With the solution of this simplified problem, we update $\tilde{u}$, thus obtaining new weights, and a new energy. We solve it again with FISTA, and iterate. Although the minimization problem to be solved within each iteration is convex, in general we cannot hope to find the global minimum of Eq. 4.16.

4.3.2.6 Multiscale Image Sampling

In the work from Buehler *et al.* (2001), they point out that although a value of $|\det D\beta_i| > 1$ can lead to oversampling artifacts (e.g, aliasing), the use of mip-mapping avoids the need to penalize images for oversampling. As a consequence they propose to lower-threshold their resolution penalty to 0. With our equations, this action is equivalent to upper-threshold our weight term $|\det D\beta_i|$ to 1. Let us study the oversampling problem, the existing solutions and its consequences on the energy term.

Supersampling techniques Supersampling (Heckbert, 1989) is the technique of minimizing the distortion artifacts, known as aliasing, when representing a high-resolution image at a lower resolution. Anti-aliasing means removing signal components that have a high frequency that can not be properly preserved by the new sampling rate. This removal is done before the subsampling at a lower resolution and is known as the *prefilter* step. In Fig. 4.6 we reproduce the original figure from Heckbert (1989) illustrating the ideal resampling process for a one dimensional signal. If the prefilter step is skipped, noticeable artifacts arise, as samples "randomly" select high-frequencies of the warped input $g_c(x)$ (see Fig. 4.5, *Nearest Neighbor*).

Several methods have been proposed to reduce those artifacts. Trilinear Mip-mapping (as implemented by OpenGL) is a commonly used prefilter technique, where the filter is isotropic (i.e. the scaling factor is the same along two directions). Mip-mapping is only mathematically accurate in the case where the transformation β_i is an isotropic scale factor, which is, in our case, usually not true (see Fig. 4.5 *Mipmaping*). McCormack *et al.* (1999) proposed an anisotropic prefilter method named *Feline*, being 4-5 times slower than mipmapping, but producing less artifacts. Other examples of supersampling anisotropic prefilter methods are *Ripmaps* or *Summed-Area Table*, illustrated in Fig. 4.5. The best results should be obtained with EWA (Heckbert, 1989), but its computational cost is about 20 times higher than mipmapping.

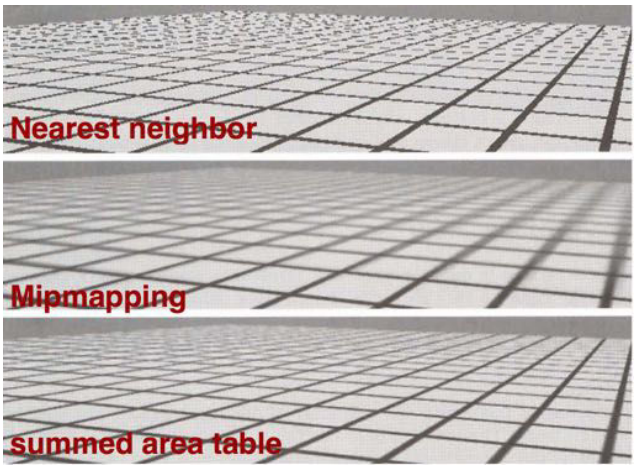


Fig. 4.5: *A texture is warped with a slanted plane. The resulting warp is highly anisotropic as the image is only compressed in the vertical direction. The images present three examples of texture mapping and its resulting artifacts. Nearest Neighbor does not supersample and fails to preserve the straight lines in the top of the image, thus creating artifacts know as black-and-white noise. Mipmapping is an example of an isotropic supersampling filter. Because the warp is not isotropic, the resulting image has important blur artifacts. Summed area table is an example of an anisotropic supersampling filter, better suited for this warp. The resulting image has fewer artifacts. Figure reproduced from Akenine-Möller* et al. *(2008).*

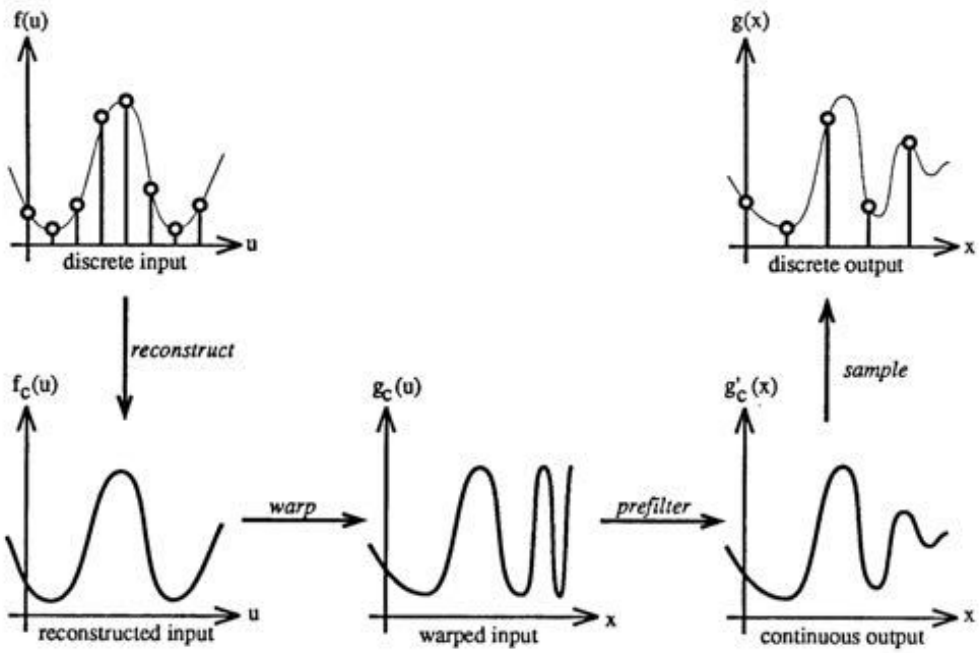


Fig. 4.6: *A discrete input $f(u)$ is reconstructed as the continuous function $f_c(u)$, warped into $g_c(x)$ -typo in the figure-, prefiltered into $g'_c(x)$ and sampled into the discrete output $g(x)$. Figure reproduced from Heckbert (1989).*

Impact of supersampling on the warped input images The weight $|\det D\beta_i|$ in equation 4.21 was devised using the assumption that u is a continuous function. In practice, we use a discrete version u^d of u for computation. In the original paper by Wanner and Goldluecke (2012), u is assumed to be a high-resolution super-resolved image with respect to v_i. However, especially for generic camera configurations, it may occur that the transformation τ_i from v_i to u^d compresses several pixels from v_i onto one discrete pixel in u^d. At these places in the image, u^d is not super-resolved with respect to v_i and τ_i, but under-resolved. Although $|\det D\beta_i| = |\det D\tau_i|^{-1}$ may become very large, we claim that, because of the prefilter step in the supersampling process, there is no reason to give more weight to these pixels.

Let us assume that we have a higher-resolution version of v_i, that we name v_i'. Because v_i' is more resolved than v_i, the warp β_i' warping v_i' into u is so that $|\det D\beta_i'| > |\det D\beta_i|$. As we can see in Fig. 4.7, v_i' only provides more frequency information than v_i at locations where $|\det D\beta_i < 1|$. Warped values where $|\det D\beta_i| > 1$ do not provide more information. We thus chose to modify the weight $|\det D\beta_i|$ whenever compression occurs. For a one-dimensional transform, this would be done by thresholding the weight, so that it is less than or equal to 1. For a two-dimensional transform, there may be an expansion along one direction, and a compression along the other. To consider this phenomena we compute the singular value decomposition (SVD) of $D\beta_i$ as $D\beta_i = U\Sigma V^*$, where U and V are orthogonal matrices, and Σ is a diagonal matrix with the singular values s_1 and s_2 on the diagonal. Each of these values corresponds to the scaling performed by $D\beta_i$ on orthogonal directions. Any scaling larger than 1 means that u^d is under-resolved in that direction, and we thus recompute the weight as the product of the thresholded singular values:

$$|\det D\beta_i|' = \min(1, s_1)\min(1, s_2). \tag{4.23}$$

Note that since $D\beta_i$ is a 2×2 matrix, the singular values can easily be computed in closed-form using the "direct two-angle method".

Impact of supersampling on warped sensor noise The supersampling process is also applied to the term $\omega_i(u)$ from Eq. 4.18 as it is composed with the function β_i in Eq. 4.21, in order to be evaluated at Γ. Let us study the impact on both sensor noise σ_s and σ_g.

In Fig. 4.8 we reuse the scheme of Heckbert (1989) and add the error bars to illustrate how an independent identically distributed error is affected by the warp. We see that in the prefilter step, areas of the signal which have been compressed contain an attenuated error, whereas, in areas of the signal which have been expanded the error is unchanged. The intuitive idea behind this phenomenon is that if a pixel in u^d is computed as the combination of several pixels in v_i, each having a Gaussian independent sensor noise, the more measures contribute to the final pixel in u the less noisy the final estimate should be. In this case, having an image v_i' with a higher resolution than v_i, translates into a larger prefilter kernel b'

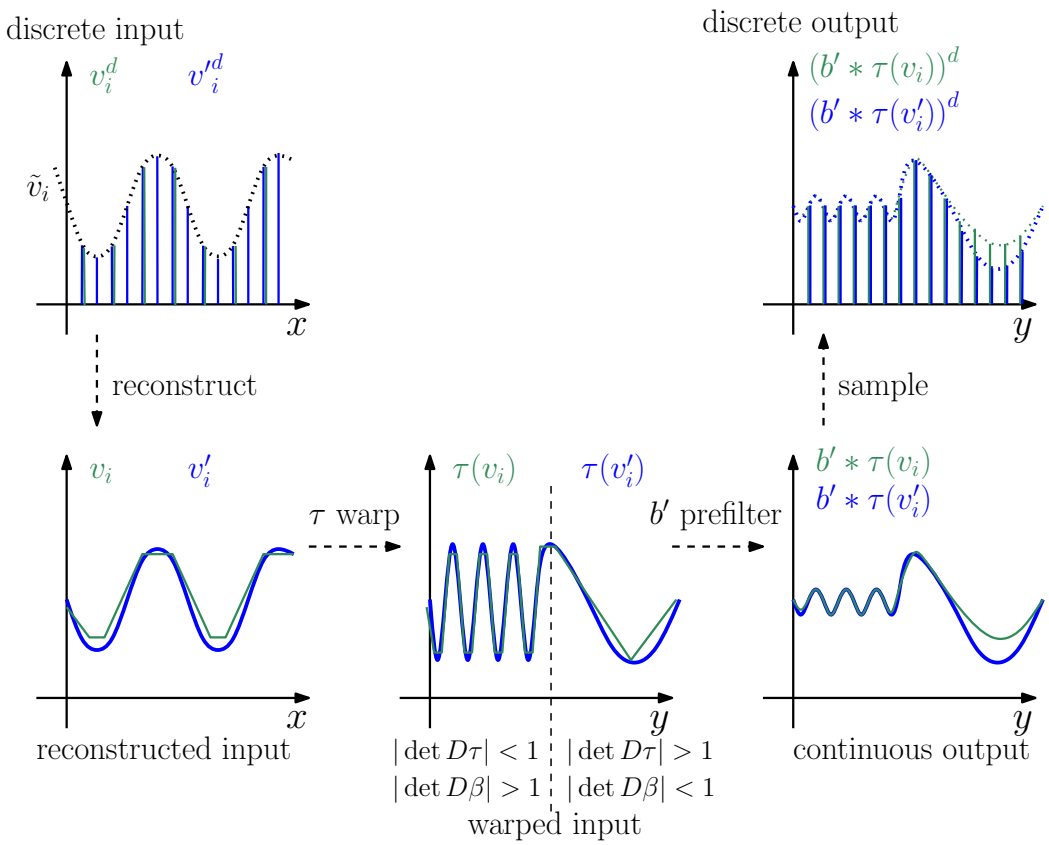


Fig. 4.7: *A signal $\tilde{v}_i$ is sampled with two different sampling rates: v_i^d and v'^d_i. The sampling rate of v'^d_i is higher than v_i^d. The discrete input is reconstructed, warped, prefiltered and sampled. The difference in sampling only creates differences in the discrete output at locations where the warp expands the signal ($|\det D\tau| > 1$ or $|\det D\beta| < 1$).*

and thus a higher reduction of the error.

When $|\det D\beta_i| > 1$, the error in the image is proportionally reduced by the supersampling factor $|\det D\beta_i|$. For a one-dimensional transform, this would be done by dividing the warped error with the supersampling factor. For a two-dimensional transform, we reuse the singular values s_1 and s_2 on the diagonal of the SVD decomposition of the $D\beta_i$ to compute the warp of the error σ^2:

$$\sigma^2 \circ \beta_i = \frac{\sigma^2}{|\det D\beta_i|''}, \tag{4.24}$$

where

$$|\det D\beta_i|'' = \max(1, s_1)\max(1, s_2). \tag{4.25}$$

The proposed reasoning is valid for the sensor noise σ_s, defined at the pixel level on the images v_i. However, the error ε_{g_i} (Eq. 4.9) associated with σ_g, arises from an error in the warp. It is yet unclear how this warp error is affected by the supersampling method.

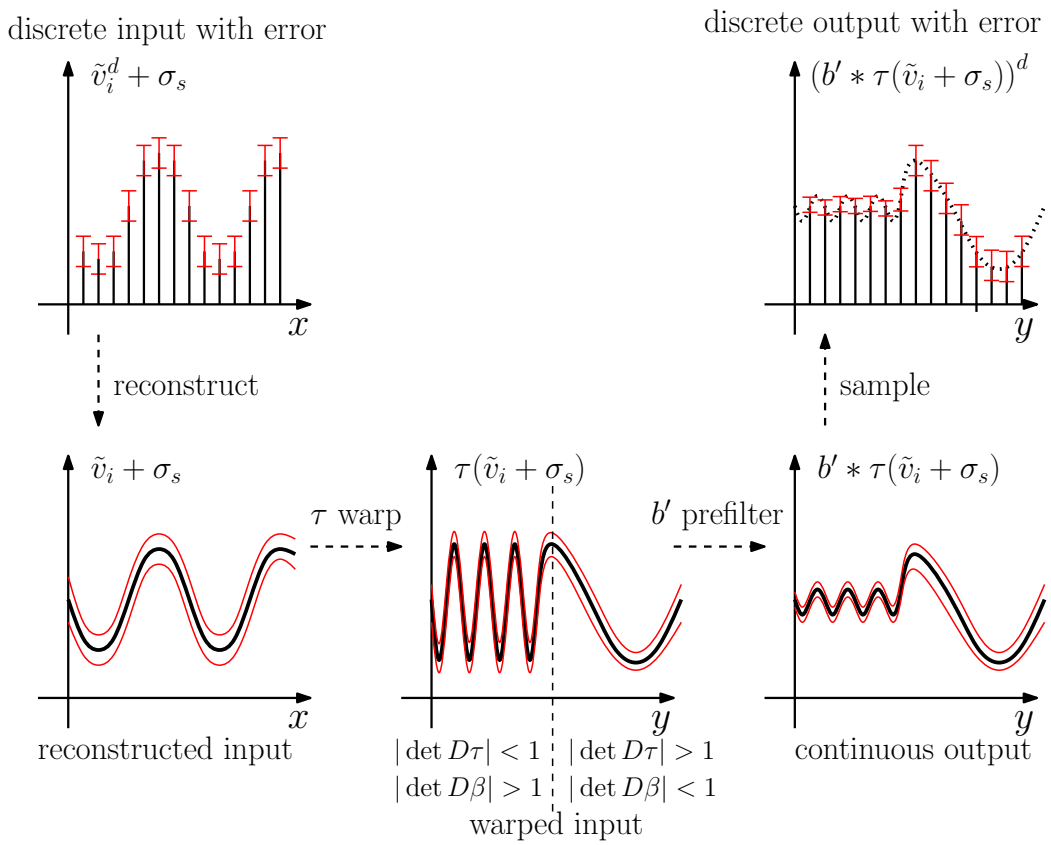


Fig. 4.8: *A discrete input with error ($\tilde{v}_i^d + \sigma_s$) represented by a sinusoidal signal is warped with a function τ. First we reconstruct the continuous input from the discrete samples with error. Then we warp the reconstructed input. Our warp compresses part of the signal and expands another part. The prefilter step b' filters out the high-frequencies and preserves the low-frequencies. The error is thus reduced where the signal is compressed, and stays unmodified where the signal is expanded.*

In our image formation model, the error ε_{g_i} is defined as the difference between the u values warped with the perfect warp $\tau*$, and the u values warped with the estimated warp τ. By definition, the error ε_{g_i} is equal for all pixels v_i that are warped into the same u location. This error cannot be represented anymore with error bars as we did in Fig. 4.8, because the error ε_{g_i} corresponds in fact to a systematic vertical shift of the warped signal. Because the warped values under the prefilter kernel b' have the same systematic shift, the resulting prefiltered signal still contains the same systematic shift. The error ε_{g_i} associated with σ_g is thus unaffected by the supersampling method.

Final energy with consideration of the supersampling process With the consideration of the weights modifications introduced by the supersampling process, the data term of the energy from Eq. 4.21 can be then rewritten as

$$E_{\mathrm{d}}(u) = \sum_{i=1}^{n} \frac{1}{2} \int_{\Gamma} \frac{|\det D\beta_i|}{\sigma_s^2 + |\det D\beta_i|''(\sigma_{g_i}^2 \circ \beta_i)} \left(m_i \left(b * (u \circ \tau_i) - v_i\right)^2\right) \circ \beta \, dx \quad (4.26)$$

where we use the fact that

$$|\det D\beta|'|\det D\beta|'' = |\det D\beta|, \tag{4.27}$$

as min and max from Eqs. 4.23 and 4.25 cancel out.

Let us point out that if $\sigma_{g_i} = 0$, the obtained energy is equal to the one proposed by Wanner and Goldluecke (2012). In their case, even with the proper consideration of the supersampling, there is no reason to threshold $|\det D\beta|$, as the supersampling factor introduced by the foreshortening effects is compensated by the reduction of the sensor noise.

In addition, let us recall that Buehler *et al.* (2001) proposed to threshold the resolution penalty to zero, because they claimed that there is no need to penalize images for oversampling. Indeed, we have shown that there is no reason to penalize them. Moreover, if one considers the sensor noise, they should be (marginally) preferred over an equal resolution image, because the supersampled sensor noise is smaller. This subtle detail was overseen in Buehler *et al.* (2001).

Moreover, let us also recall that Kopf *et al.* (2014) proposed to penalize the *resolution sensitivity* based on the ratio between the minimal and maximal singular values, instead of the determinant of the Jacobian of the warp. They observed, that the determinant could be small even for regions with an important stretch of the image and proposed an heuristic to counter these undesirable effects. Their proposed heuristic does not penalize images with a higher resolution.

4.4 Simplified Camera Configuration Experiments

In order to evaluate the proposed approach we proceed in two stages. First we conduct experiments in a simplified camera configuration. This configuration is chosen so that the equations are simplified and allows us to validate a simplified implementation of the optimization procedure. In the next section we consider the fully general case, where camera poses are unconstrained. For both configurations we perform experiments with both synthetic and real-world scenes. The synthetic datasets allow us to validate our approach with ground truth information. The real-world scenes allow to state that the method is also valid for actual images.

We conduct a first set of experiments in a simplified camera configuration. This allows us to use a simplified implementation of the optimization procedure. In this set of experiments we suppose that our cameras have a simplified configuration. Specifically, all viewpoints are in a common plane, which is parallel to all image planes, i.e. we are dealing with a 4D light field in the Lumigraph parametrization (Gortler *et al.*, 1996). The novel view is also synthesized in the same image plane, which means that τ_i is simply given by a translation proportional to the normalized disparity d_i,

$$\tau_i(x) = x + d_i(x)(c - c_i). \tag{4.28}$$

The normalized disparity is expressed in pixels per world units, and is together

with its associated uncertainty related to depth via:

$$d_i(x) = \frac{f_i}{z_i(x)} \text{ and } \sigma_{d_i}(x) = \sigma_{z_i}(x)\frac{f_i}{z_i(x)^2}, \tag{4.29}$$

where f_i is the camera focal length expressed in pixels.

Plugging Eq. 4.29 and Eq. 4.28 into Eq. 4.15, we derive the link between the geometric error and its associated image error as:

$$\sigma_{g_i} = \sigma_{d_i} \left|(b * ((\nabla u \circ \tau_i) \cdot (c - c_i)))\right|, \tag{4.30}$$

where σ_{d_i} models the disparity noise. Finally, the deformation term in Eq. 4.22 is

$$|\det D\beta_i| = |\det D\tau_i|^{-1} = |1 + \nabla d_i \cdot (c - c_i)|^{-1}. \tag{4.31}$$

4.4.1 Structured Light Field Datasets

To validate the theoretical contribution, we compare results on two light field datasets: The HCI Light Field Database Wanner *et al.* (2013), and the Stanford Light Field Archive Vaish and Adams (2008). These datasets provide a wide collection of challenging synthetic and real-world scenes.

In a first set of experiments, we render an existing view from the dataset at the same resolution, without using the respective view as an input to the algorithm. We consider two different qualities of geometric proxy: an approximate one from estimated disparity maps (Wanner and Goldluecke, 2014), and an extremely poor one represented by an infinite flat fronto-parallel plane in the estimated center of the scene. We adapt σ_{d_i} accordingly, i.e. when using the estimated disparity, we use a value corresponding to the expected accuracy of the reconstruction method: $\sigma_{d_i} = \frac{d_{\max} - d_{\min}}{\text{nbLayers}}$, where nbLayers is the number of disparities considered by the method. When a bare plane in the middle of the scene is used, we instead use $\sigma_{d_i} = \frac{d_{\max} - d_{\min}}{4}$. In all cases, $\sigma_s = 1/255$.

A second set of experiments is performed by rendering a 3×3 super-resolved image from a set of 5×5 input views. Although super-resolution is not the main purpose of this work, we also provide a comparison with the state of the art. As super-resolution relies on sub-pixel disparity values, using a plane as the geometric proxy has little interest. We only show the results obtained with the estimated disparity maps.

4.4.2 Numerical Evaluation

In Tab. 4.1, we show the numerical results obtained by our method, and compare them to the ones achieved with Wanner and Goldluecke (2012). We use two state of the art image quality full reference measures. The Peak Signal to Noise Ratio (PSNR), which computes a value in dB units. The bigger the dB value, the better the generated signal is. We also use the Structural SIMilarity (SSIM) metric (Wang

	HCI light fields, raytraced				HCI light fields, gantry				Stanford light fields, gantry					
	still life		*buddha*		*maria*		*couple*		*truck*		*gum nuts*		*tarot*	
Estimated disparity														
SAVS	30.13	58	**42.84**	**17**	40.06	**53**	26.55	226	33.75	408	31.82	1439	28.71	60
Proposed	**30.67**	**52**	42.45	**17**	**40.13**	**53**	**28.53**	**178**	**33.79**	**406**	**31.99**	**1435**	**28.98**	**57**
Planar disparity														
SAVS	21.28	430	34.28	74	31.65	144	20.07	725	32.48	419	30.55	1403	22.64	278
Proposed	**22.24**	**380**	**37.51**	**44**	**34.38**	**99**	**22.88**	**457**	**33.79**	**386**	**31.30**	**1378**	**23.78**	**218**
Super-resolution														
SAVS	24.93	230	**34.50**	**122**	35.18	129	**25.54**	**287**	**33.11**	**378**	31.80	1475	**26.66**	**113**
Proposed	**25.21**	**224**	34.44	123	**35.23**	**128**	25.37	288	33.10	**378**	**31.93**	**1462**	26.67	114

Table 4.1: *Numerical results for synthetic and real-world light fields from two different online archives. We compare our method (Proposed) to Wanner and Goldluecke (2012) (SAVS) with respect to same-resolution view synthesis for estimated disparity and a flat plane proxy, as well as super-resolved view synthesis. For each light field, the first value is the PSNR (bigger is better), the second value is DSSIM in units of* 10^{-4} *(smaller is better). The best value is highlighted in bold. See text for a detailed description of the experiments.*

et al., 2004), which was developed to be more consistent with human eye perception. Whereas PSNR relies on a per pixel local computation, SSIM considers the image structure with the help of local windows. We report results with the distance DSSIM based on SSIM:

$$\mathrm{DSSIM} = \frac{1 - \mathrm{SSIM}}{2}, \tag{4.32}$$

which has no units. The smaller the DSSIM value, the more similar both images are. For our comparison we measure the PSNR and DSSIM values between the actual and generated images. Although our method visibly performs better, numerical values should be interpreted carefully. In Fig. 4.9 we show detailed closeups illustrating the benefits or our method. As high resolution images are not available for most of the datasets, PSNR and DSSIM values for the super-resolved images are computed by subsampling the input images, generating the novel super-resolved view and comparing it with the original one.

When rendering with precise geometry, both methods are roughly equivalent with respect to PSNR and DSSIM values. These values are presented in Tab. 4.1 in the rows *Estimated disparity* and *Super-resolution*. When the quality of the proxy degrades, our method clearly outperforms previous work, taking advantage of the explicit modeling of depth uncertainty. These values are presented in Tab. 4.1 in the rows *Planar disparity*. As shown in the closeups of Fig. 4.9, our method better reconstructs color edges in all configurations. Full-resolution images are provided in the Appendix B.

4.4.3 Processing Time

Computation time when rendering at target resolution 768×768 with 8 input images is on the order of 2 to 3 seconds. Computation time for super-resolved view synthesis with a factor of 3×3 and 24 input images is around 2 to 3 minutes. All experiments used an nVidia GTX Titan GPU.

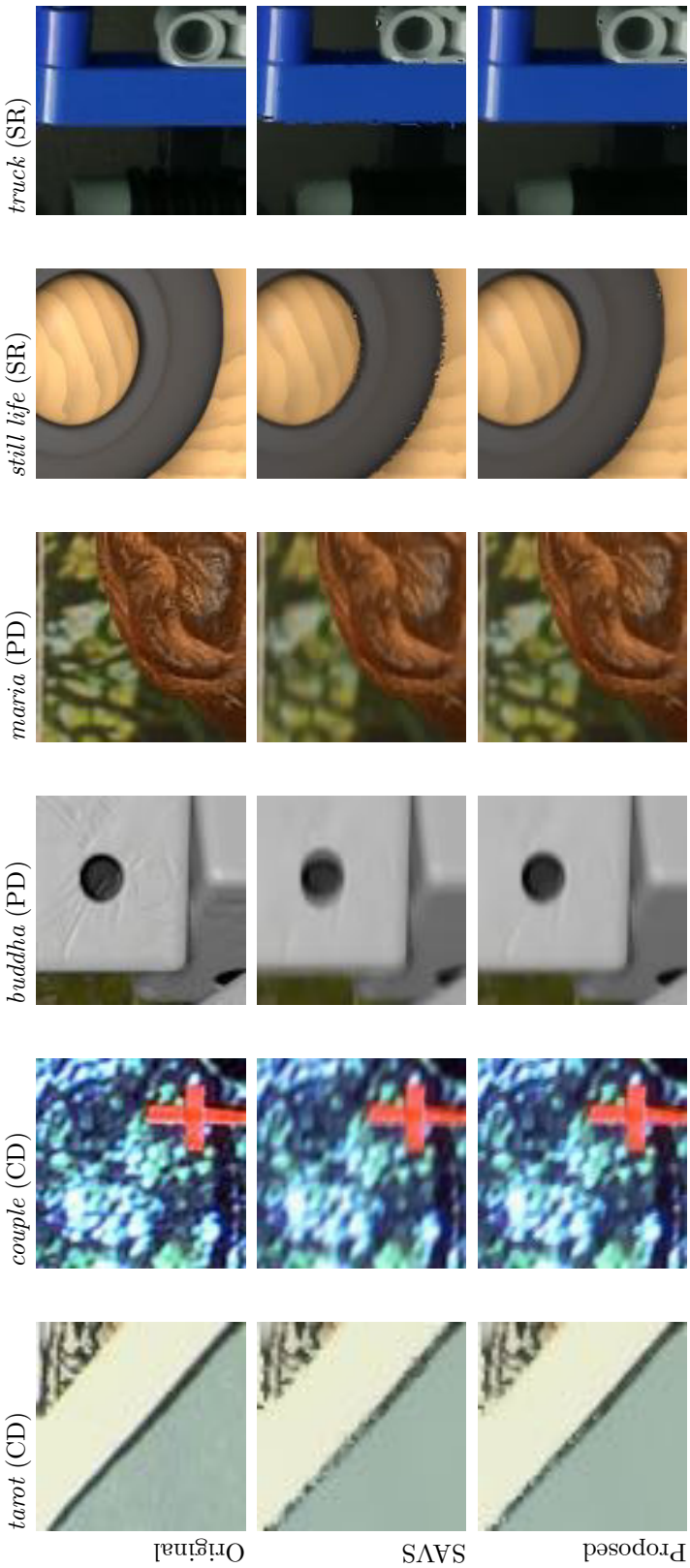


Fig. 4.9: *Visual comparison of novel views obtained for different light fields. From top to bottom, the rows present closeups of the ground truth images (Original), the results obtained by Wanner and Goldluecke (2012) (SAVS), and our results (Proposed). CD stands for computed disparity, PD for planar disparity and SR for super-resolution, see text for details. Full resolution images can be found in the Appendix B. The results obtained by the proposed method are visibly sharper, in particular along color edges.*

4.5 Experiments on Generic Camera Configuration

As stated by Wanner and Goldluecke (2012): *an implementation in this generality would be quite difficult to achieve. We, (...) leave a generalization of the implementation for future work.*

In this section we detail the implementation of the proposed model in the generic camera configuration. In the first subsection we detail the input generation, starting with a set of input images, and obtaining a 3D reconstruction with its associated uncertainty. In the second part we derive the equations of the transfer functions and weights in general form for an unstructured configuration. In the third section we present the datasets on which we run experiments. The fourth part presents the obtained results including a discussion and future leads.

4.5.1 Input Generation: 3D Reconstruction and Uncertainty Computation

Our algorithm needs as input a set of warping functions τ_i and their associated uncertainty. We will perform our experiments using an explicit geometric proxy. However, as our method is also capable to use an implicit geometric proxy we briefly explain in the next subsection how we would handle such an input.

4.5.1.1 Implicit Geometric Proxy from Disparity Maps

A simple example of a geometric implicit proxy is the case of viewpoint interpolation between two rectified input cameras. A geometric proxy can be computed in the form a disparity map between each pair of input cameras (d_{ij}), The warps τ_1 and τ_2 from the images to a virtual camera lying between the input cameras v_1 and v_2, at a fraction α can be computed as a fraction of the disparity between the images

$$\begin{cases} \tau_1 = \alpha d_{12}, \\ \tau_2 = (1-\alpha) d_{21}, \end{cases} \tag{4.33}$$

where d_{12} is the disparity between the camera 1 and 2 and d_{21} the disparity between the camera 2 to 1. Then all necessary magnitudes for our method are available, without the need to explicitly reconstruct the geometry.

This warp computation without an explicit geometric reconstruction can be generalized to multiple images (Chen and Williams, 1993; Laveau and Faugeras, 1994). In their seminal work, Chen and Williams (1993) first connect the source images to create a graph structure, in the form of a 3D lattice of tetrahedra. For each pair of connected images, they compute a *morph map*, describing the 2D mapping from one image to another. This concept is similar to the one dimensional disparity map, but more general as cameras do not need to be rectified. Then they use the barycentric coordinates of the target view location to interpolate among the images attached to the vertices of the enclosing tetrahedron. The main restriction of this approach is that the view location has to be inside the graph structure.

Laveau and Faugeras (1994) use the epipolar geometry to establish correspondences from the input images into a target image with an arbitrary location. They first compute image correspondences between the input images, in order to calibrate the cameras and create disparity maps. Then, with a clever use of the epipolar geometry and 3D point triangulation, they compute the warps from the input images into the target image.

Our method can work with those warps as input. As the uncertainty of those warps may not be available, in Sec. 4.5.1.3 we propose possible ways to estimate it.

4.5.1.2 Camera Calibration and 3D Reconstruction in our Experiments

For our experiments we first *calibrate* the cameras and obtain their camera matrices $\boldsymbol{P}_i$ together with their decomposition into intrinsic and extrinsic parameters $(\boldsymbol{K}_i, \boldsymbol{R}_i, \boldsymbol{t}_i)$. The multi camera calibration problem has been intensively studied (Triggs *et al.*, 2000; Hartley and Zisserman, 2004). We use the OpenMVG library (Moulon *et al.*, 2013) to calibrate the cameras and obtain a first set of correspondences. Then we use PMVS2 (Furukawa and Ponce, 2010) to extract a relatively large set of 3D points. If the size of the reconstructed scene is large, we use the CMVS algorithm (Furukawa *et al.*, 2010), which subdivides the scene in small clusters, reconstructs them using PMVS2 and then merges them together into a final set of matches. PMVS2 also provides the normal vector associated with each patch, providing orientation information of the scene. Then we use a *Poisson reconstruction* (Kazhdan *et al.*, 2006) to fit a mesh to the obtained point cloud with normals. We obtain a set of 3D points $\boldsymbol{p}_j$, and a set of triangles relying them.

4.5.1.3 Modeling Geometric Uncertainty and Depth Error

Once we have a geometrical proxy, we would like to have an estimate of its uncertainty. Unfortunately, the best state of the art algorithms in unstructured, uncontrolled scenes (Pollefeys *et al.*, 2008; Furukawa and Ponce, 2010; Gallup *et al.*, 2010) do not provide a geometric uncertainty of their estimation. In general, they provide a confidence measure in form of a score, often associated to the photo-consistency of the 3D point when projected into the images. As we saw in Sec. 4.2.2.2, this score should be used with caution, because the confidence measure from a depth estimator is unit-less, whereas the geo-uncertainty is in world units. Although tempting, one should avoid to take the score measures *as* the geometric uncertainty. However, there are other ways to obtain the geometric uncertainty. One simple way is to consider the discretization error. For example, if the reconstruction method computes 1D or 2D disparity values, and those disparities are discretized in integer values, assuming an error of ± 0.5 disparities seems acceptable. Then the warp error could be approximated by the normal distribution $\mathcal{N}(0, 0.5)$. In our case we use 3D points, which have been triangulated from the images. It seems reasonable to use the uncertainty of the 3D point in the image location to compute its geometric uncertainty.

4.5.1.4 Computing the Geometric Uncertainty of a 3D Point

Given a triangulated 3D point from matched feature points, we can use the uncertainty of the matching algorithm to compute the geometric uncertainty of the 3D point. Zeisl *et al.* (2009) compute the location uncertainty for the feature positions computed with the *SIFT* (Lowe, 2004) and *SURF* (Bay *et al.*, 2008) feature detectors. Those uncertainties are in the 2D image domain and have the form of a 2x2 covariance matrix $\boldsymbol{S}$. The features point uncertainty can be then backprojected into the 3D space. The generic formula to compute the resulting 3D uncertainty by back-projecting all the matching points in each camera can be found in Chapter 4.6 of Heuel (2004). A more specialized formula for the camera projection matrices can be found in Hartley and Zisserman (2004). Let $\boldsymbol{P}_i$ be N camera matrices in the form

$$\boldsymbol{P}_i = \begin{pmatrix} \boldsymbol{p1}_i \\ \boldsymbol{p2}_i \\ \boldsymbol{p3}_i \end{pmatrix}. \tag{4.34}$$

Let $\boldsymbol{y}_i$ be a 2 dimensional vector containing the image coordinates of a matched point in the image i. Let $\boldsymbol{S}_i$ be the 2x2 covariance matrix of the match in the i'th image. Let $\boldsymbol{x}$ be the 3D point corresponding to the matched feature and $\bar{\boldsymbol{x}}$ its homogeneous coordinate extension. Let us consider the non-linear regression problem

$$\begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_N \end{pmatrix} = \begin{pmatrix} f_1(\boldsymbol{x}) \\ f_2(\boldsymbol{x}) \\ \dots \\ f_N(\boldsymbol{x}) \end{pmatrix} + \mathcal{N}(0, \boldsymbol{S}), \tag{4.35}$$

$$\text{where} \quad f_i(\boldsymbol{x}) = \begin{pmatrix} \frac{\boldsymbol{p1}_i \cdot \bar{\boldsymbol{x}}}{\boldsymbol{p3}_i \cdot \bar{\boldsymbol{x}}} \\ \\ \frac{\boldsymbol{p2}_i \cdot \bar{\boldsymbol{x}}}{\boldsymbol{p3}_i \cdot \bar{\boldsymbol{x}}} \end{pmatrix} \quad \text{and} \quad \boldsymbol{S} = \begin{pmatrix} \boldsymbol{S}_1 & \boldsymbol{0} & \dots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{S}_2 & \dots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \dots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \dots & \boldsymbol{S}_N \end{pmatrix}. \tag{4.36}$$

The maximum likelihood estimate of $\boldsymbol{x}$ is the solution to the non-linear least squares problem:

$$\boldsymbol{x}^* = \arg\min_x ||(f(\boldsymbol{x}))^\top \boldsymbol{S}^{-1} f(\boldsymbol{x})||^2, \tag{4.37}$$

where $f(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \dots, f_N(\boldsymbol{x}))$ and the operator $^\top$ denotes the transpose of the vector. The covariance $\boldsymbol{C}$ of $\boldsymbol{x}^*$ is given by

$$\boldsymbol{C}(\boldsymbol{x}^*) = (\boldsymbol{J}^\top(\boldsymbol{x}^*)\boldsymbol{S}^{-1}\boldsymbol{J}(\boldsymbol{x}^*))^{-1}, \tag{4.38}$$

where $\boldsymbol{J}(\boldsymbol{x}^*)$ is the Jacobian of f at $\boldsymbol{x}^*$.

Per Vertex Geometric Uncertainty in our Experiments To construct the matrices of Eq. 4.38 we need to know if the vertex $\boldsymbol{p}_j$ is visible on the camera i (or not). We first compute the depth maps $z_i(\boldsymbol{x}) : \Omega_i \rightarrow \mathbb{R}$ by projecting the mesh on each view. Then, for a given vertex $\boldsymbol{p}_j$, and a camera i, we recompute its depth

$$(\boldsymbol{R}_i \boldsymbol{p}_j)[3] + \boldsymbol{t}_j[3], \tag{4.39}$$

and compare it to the value in the depth map $z_i(\boldsymbol{x})$. If both are equal (up to depth quantization noise), the vertex $\boldsymbol{p}_j$ is seen by the camera i. Otherwise, the vertex $\boldsymbol{p}_j$ is not seen in the camera i. With the set of cameras we can compute the Jacobian matrix $\boldsymbol{J}$ and its transpose $\boldsymbol{J}^\top$. To construct the matrix $\boldsymbol{S}$, in our experiments we assume a one pixel uncertainty in the image location, and so the 2D covariance matrices are the identity matrix: $\forall i\ \boldsymbol{S}_i = \boldsymbol{I}$. Note that in this process the resolution of the camera is taken into account, as the uncertainty in the images is given in pixel units. Thus when we convert pixel units into world units, a high-resolution camera has *smaller* pixels than a low-resolution camera.

4.5.1.5 Computing the Geometric Uncertainty of a 3D Mesh

Once we have this per-vertex covariance we compute the geometric uncertainty in the surface mesh. We propose to focus on the uncertainty along the direction of the normal of the surface. The idea is that, if the surface is smooth, a variation of the vertex position *on* the surface does not affect much the shape of the surface. We illustrate this idea in Fig. 4.10. This allows us to reduce the dimensionality of the covariance matrix to 1, by only considering the uncertainty along the normal vector. We compute the geometric uncertainty σ_n of the vertex $\boldsymbol{p}$ in the direction of the normal $\boldsymbol{n}$ by projecting the covariance matrix $\boldsymbol{C}$ onto the normal direction:

$$\sigma_n(\boldsymbol{p}, \boldsymbol{n}, \boldsymbol{C}) = \boldsymbol{n}^t \boldsymbol{C} \boldsymbol{n}. \tag{4.40}$$

Then, the problem of interpolating the uncertainty on the mesh surface is reduced to a scalar interpolation (σ_n) plus a normal vector interpolation ($\boldsymbol{n}$).

The correct interpolation of the normal vector demands some attention. In our implementation we perform an approximation with a linear interpolation of the normal vectors. Given two vectors $\boldsymbol{n_0}$ and $\boldsymbol{n_1}$, their linear interpolation is

$$\hat{\boldsymbol{n}}_{\boldsymbol{\alpha}} = (1 - \alpha)\boldsymbol{n_0} + \alpha \boldsymbol{n_1}. \tag{4.41}$$

The obtained vector $\hat{\boldsymbol{n}}_{\boldsymbol{\alpha}}$ does not have a unit norm, so we have to normalize it in order to obtain valid normal vector $\boldsymbol{n}_{\boldsymbol{\alpha}} = \frac{\hat{\boldsymbol{n}}_{\boldsymbol{\alpha}}}{||\hat{\boldsymbol{n}}_{\boldsymbol{\alpha}}||}$. This interpolation is not correct because we are interpolating the chord in the circle, instead of the angle between the vectors. However, the error is small for small angles and its computation is very fast. The well known lighting technique of *Phong Shading* (Phong, 1975) does the same approximation.

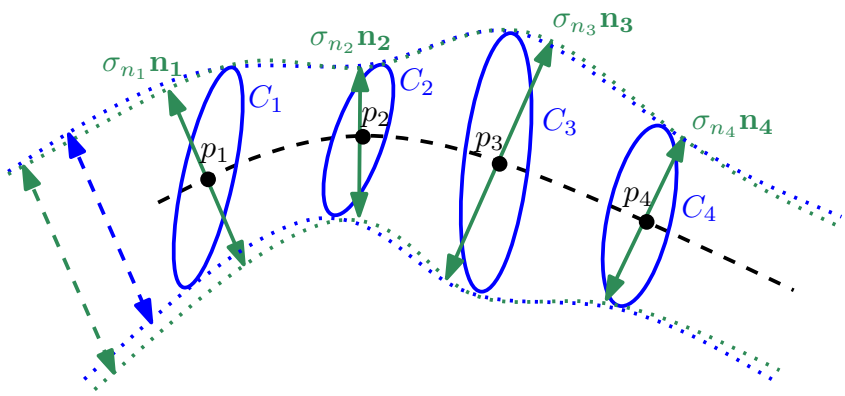


Fig. 4.10: *A set of vertices $\boldsymbol{p_j}$, their covariance matrices $\boldsymbol{C_j}$, and their normal vector $\boldsymbol{n_j}$ scaled with σ_{n_j}. The global uncertainty computed with the covariance matrices (in blue), is very similar to the one obtained with the computed covariance $\sigma_{n_j}\boldsymbol{n_j}$ (in green). We use the sub-index j to refer a vertex. Not to be confused with i, which we use to refer the input view index.*

4.5.1.6 Mapping the 3D Geometric Uncertainty to Depth Uncertainty in the Images

The last step in the input creation is to obtain the per-pixel depth uncertainty $\sigma_{z_i} : \Omega_i \rightarrow \mathbb{R}$ for each view i. We propose two methods. The first performs computations locally. It allows a fast parallel computation, as each pixel can be treated independently, but disregards global effects, e.g. potential occlusions of other parts of the mesh. The second approach is global and takes into account the full mesh in the computations. While more accurate, the computational cost of the second method is approximately the double, as we need to render the 3D mesh twice.

Local computation Let $\boldsymbol{p}$ be the first intersection between the viewing ray corresponding to the pixel $\boldsymbol{x}$ and the 3D mesh, as illustrated in Fig. 4.11. Together with the vertex $\boldsymbol{p}$ we obtain its (possibly interpolated) normal vector $\boldsymbol{n}$ and the geometric uncertainty along the normal vector direction σ_n. If we consider the mesh to be locally planar (Fig. 4.11a), the depth uncertainty *as seen* from the viewpoint i can be computed using the angle α between the normal and the viewing ray. We note the explicit dependence with $\alpha(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{n})$. Then the pixel uncertainty at $\boldsymbol{x}$ in the view i is

$$\sigma_{z_i}(\boldsymbol{x}) = \frac{\sigma_n}{\sin(\alpha(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{n}))}. \tag{4.42}$$

Fig. 4.11b illustrates a limitation of this approximation, when the geometry of the mesh is not planar. The interactions between the different vertices are not captured by the local approximation of the geometric uncertainty.

Global computation: the Geometric Variations In order to take into account the global geometry we propose to use what we name *geometric variations*. We consider the computed mesh to be the initial mean mesh. Then we propose to

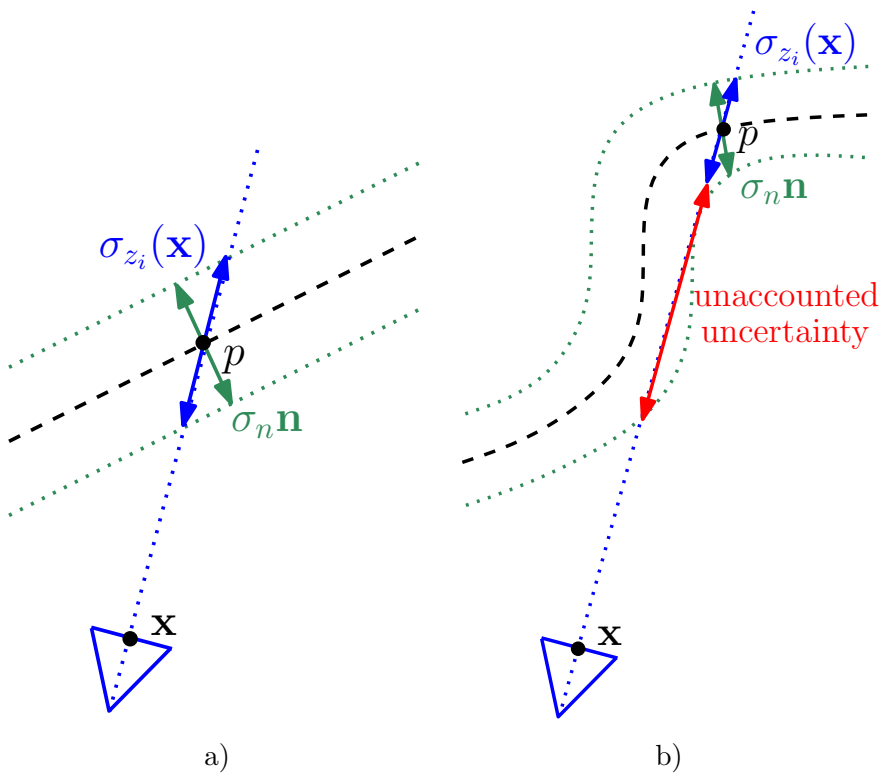


Fig. 4.11: *a) Locally approximating the surface with a planar proxy. The per view uncertainty depends on the angle between the viewing ray and the normal. b) A failure case where the local approximation does not capture the global uncertainty of the mesh.*

inflate/deflate the mesh by translating each vertex along the direction defined by its normal vector. The translation amount is the product of an inflate/deflate scalar $\delta \in \mathbb{R}$ with the geometric uncertainty of the vertex σ_{z_j}. The new set of vertices $\boldsymbol{v}_j^\delta$ is defined as

$$\boldsymbol{v}_j^\delta = \boldsymbol{v}_j + \delta\sigma_{z_j}\boldsymbol{n}_j. \tag{4.43}$$

The edges defining the neighbors of the vertices are not modified.

As we considered the depth uncertainty to be Gaussian along the normal direction, we can restrain δ to a small interval (δ^-, δ^+). For example taking $\delta \in (-2.57, +2.57)$ allows us to create a volume with a 99% probability to contain all the actual vertices. The size of the δ domain is arbitrary and can be chosen by fixing a probability value. However, the obtained pixel uncertainty varies depending on the probability value. We will discuss later in the section the choice of this parameter.

Let us now compute the geometric variations of the mesh with δ^- and δ^+. For each of both new geometries we can compute the depth maps for each input view $z_i^\delta : \Omega_i \to \mathbb{R}$. Then, for each pixel $\boldsymbol{x} \in \Omega_i$, the difference between the computed

depths can be used as an estimator of the pixel uncertainty

$$\sigma_{z_i}(\boldsymbol{x}) = \frac{|z_i^{\delta^-}(\boldsymbol{x}) - z_i^{\delta^+}(\boldsymbol{x})|}{(\delta^+ - \delta^-)}. \tag{4.44}$$

Note that the variation of a mesh by moving the vertices may create topological problems, e.g. some triangles orientation may be reversed or some holes in the mesh may disappear. Even though we use small values of δ in the vertex displacements, those artifacts can (and do) arise. Our goal is to compute a volume in which the surface elements are with a high probability, by differentiating the depth maps of both variations. The topological artifacts are not a problem to the computation of this difference.

With this global computation, the problem illustrated in Fig. 4.11b is taken into account. The uncertainty of a vertex is affected by the neighboring vertices. Let us remark that pixels near an occlusion border have a large depth uncertainty, as depth values switch from front to back depths.

From this last remark it is obvious that even if the geometric uncertainty is Gaussian, the obtained depth uncertainty along the viewing ray is no longer Gaussian. The computed 3D covariance matrix $\boldsymbol{C}$ is a Gaussian approximation of the uncertainty. Its projection along the normal vector $\boldsymbol{n}$ is therefore also Gaussian. Moreover, the per pixel uncertainty computed with the planar approximation of the mesh, is still Gaussian. However, in the general case, the *global pixel uncertainty* computed with the geometric variations is *not Gaussian* anymore. We are fully aware of it, but for the sake of simplicity of the generative model, in the rest of this work we continue considering this uncertainty as if it was Gaussian.

Filtering the Depth Uncertainty in the Images The mesh vertices computed by the Poisson reconstruction (Kazhdan *et al.*, 2006), are not necessarily the same as the PMVS2 vertices (Furukawa and Ponce, 2010). In general, there are more mesh vertices than PMVS2 vertices. Ideally, the reconstruction uncertainty should be computed on the PMVS2 vertices, but propagating the uncertainty from vertices to a mesh is not easy. This is why we compute the uncertainty directly on the mesh vertices. However, when we compute the depth uncertainty in the images, we would like to know if the computed vertex uncertainty is likely to be from a PMVS2 vertex or not. In other words, if the vertex was "invented" by the Poisson reconstruction we know its geometric uncertainty is high. Our computed geometric uncertainty (Sec. 4.5.1.4) should not be used for those vertices. To do so we filter the depth uncertainty values with a mask $M : \Omega_i \to [0, 1]$. We first project the PMVS2 vertices to the images, and obtain a binary mask: the pixel value is zero if no PMVS2 vertex projects *near* it; the pixel value is one if at least one PMVS2 vertex projects *near* it. A binary mask example is shown in Fig. 4.12c. The *near* value could be deduced from the size of the PMVS2 vertex patch, but this information was not available when we conducted the experiments, so we used a fixed threshold. Of course it would be possible to create M as a smooth mask, by assigning to each pixel a value inversely proportional to its distance to a PMVS2 vertex projection. To create

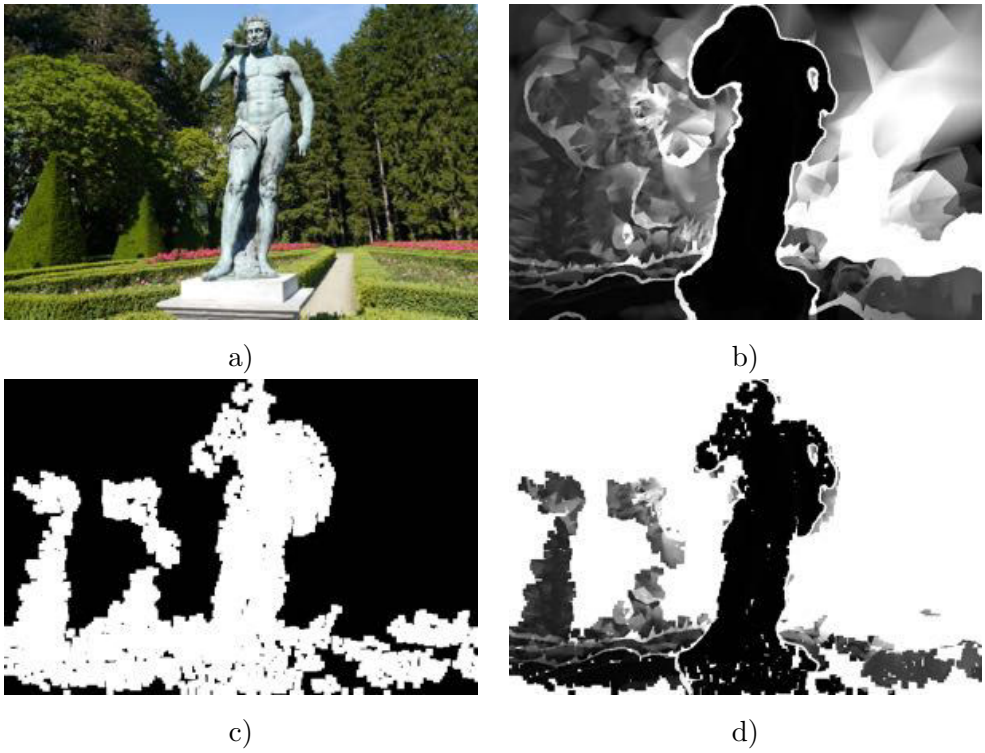


Fig. 4.12: *a) original image. b) computed depth uncertainty in the images, low uncertainty is dark (the statue), high uncertainty is bright (the trees on the right). Note the high uncertainty values on the depth discontinuities of the statue. c) PMVS2 vertices projections in the images. d) filtered depth uncertainty in the images.*

the filtered depth uncertainty in the images $\tilde{\sigma}_{z_i}$, we filter the depth uncertainty in the images σ_{z_i} from Eq. 4.44 with the binary (or smooth) mask. Pixels which are not *near* a PMVS2 vertex projection, are assigned a high uncertainty value σ_{max}. Pixels *near* a PMVS2 vertex projection keep the computed value. Moreover, for valid values of the mask, we also threshold the σ_{z_i} to σ_{max}, as we do not want to penalize a plausible geometry more than an "invented" one.

$$\tilde{\sigma}_{z_i} = \min\left(\sigma_{z_i}, \sigma_{max}\right) M + \sigma_{max}\left(1 - M\right). \tag{4.45}$$

If the mask M is smooth we perform a linear interpolation and if the mask is binary, we threshold the values. Fig. 4.12 illustrates the filtering process.

Relation between Depth Uncertainty in the Images and the Continuity desirable property The proposed method in Sec 4.5.1.4 to compute the depth uncertainty *as seen* from the input view, provides an insight into the *continuity* desirable property (Buehler *et al.*, 2001). Let us recall the property description: "*the contribution due to any particular camera should fall to zero ... as one approaches a part of a surface that is not seen by a camera due to visibility.*" As we saw, pixels near an occlusion border have a higher uncertainty due to the difference in the δ^-

depth map and the δ^+ depth map. Thus the corresponding weight in the final image (Eq. 4.18) is very small. However the transition between the occlusion area and the visible area is not smooth, because we computed the difference of the depth maps between only two variations. A smooth depth uncertainty can be achieved by performing multiple geometric variations and doing a weighted sum of the obtained results:

$$\hat{\sigma}_{z_i} = \frac{\int_{\delta^-}^{\delta^+} p(\delta)\sigma_{z_i}^{\delta} d\delta}{\int_{\delta^-}^{\delta^+} p(\delta) d\delta}, \tag{4.46}$$

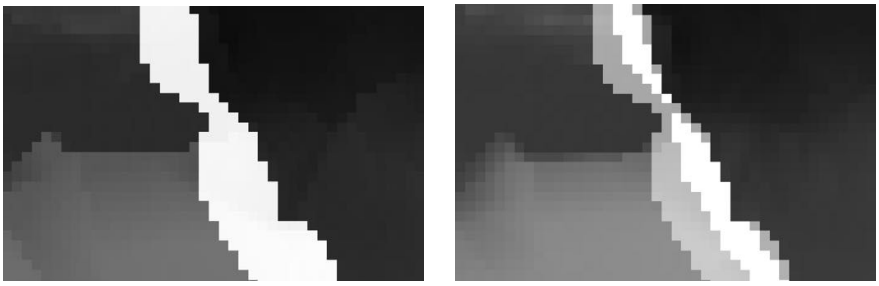
where $\sigma_{z_i}^{\delta}$ is given by Eq. 4.44 and $p(\delta)$ is a probability density function given by a normal distribution $\mathcal{N}(0,1)$. In practice, we have to choose a discretization step to approximate the integral with a finite sum as well as the limits of the integral. The higher the discretization, the smoother the transition between occlusion and visible areas is, as we illustrate in Fig. 4.13. The computational cost linearly depends on the number of discretizations used for the computation, and the computation only needs to be done once, as we only need to store the per pixel geometric uncertainty.

Note that other authors enforce this *disocclusion penalty* by setting a double threshold (Buehler *et al.*, 2001; Raskar and Low, 2002; Takahashi and Naemura, 2012). A first threshold allows to classify a depth change in the images as a discontinuity edge. A second threshold establishes the allowed maximal distance T, in pixel units, between a pixel and the detected discontinuity edge in the image. Pixels closer than this distance from the edge are penalized. Note that when using multiple images, the parameter T should be adapted depending on the view. For example, if the images are not at the same distance from the geometric element creating the depth discontinuity, the distance T should be higher for cameras closer to the geometry, and lower for cameras farther away. This consideration is automatically taken into account by Eq. 4.46.

In our method, we neither need a threshold to classify a depth change as discontinuity nor a maximal distance threshold T. Instead, we have to set the volume probability driving δ^- and δ^+ as well as a discretization step, which are the parameters needed to approximate the continuous function of Eq. 4.46. In our experiments we use a 99% volume probability and 6 geometric variations. Moreover, if one only wants to penalize pixels in occluded regions, as proposed by Raskar and Low (2002), the geometric variations can be done by setting $\delta^- = 0$ in Eq. 4.46. By tuning the volume probability and the number of discretization steps the *continuity* desirable property can be adjusted.

Let us clarify that in order to enforce the *continuity* desirable property, the filtering stage should be performed with a smooth mask $M(\boldsymbol{x})$. With a binary mask, the continuity created with the geometric variations would be broken.

Let us also note that although we found an insight into the *continuity* desirable property near the occlusion borders, we do not have yet any evidence enforcing the continuity near the borders of the image. In Sec. 4.5.2.5 we will discuss this



a) b)

Fig. 4.13: *Detail of the depth uncertainty in the images computed with a) 2 geometric variations b) 6 geometric variations. Near an occlusion border, the uncertainty is smoother as we increase the number of geometric variations to compute it. As a consequence, the weights from Eq. 4.18 smoothly fall to zero as pixels approach an occlusion border. The "continuity" desirable property is fulfilled.*

phenomenon.

4.5.1.7 Closure on input generation

Let us summarize the generation of the input for our algorithm, starting from the input images:

1. Calibrate the cameras using the input images (Moulon *et al.*, 2013): $\boldsymbol{P}_i, \boldsymbol{K}_i, \boldsymbol{R}_i$ and $\boldsymbol{t}_i$.

2. Estimate a 3D point cloud (PMVS2/CMVS) (Furukawa *et al.*, 2010; Furukawa and Ponce, 2010).

3. Estimate a 3D mesh (Kazhdan *et al.*, 2006): $\boldsymbol{p}_j$, $\boldsymbol{n}_j$ and triangles.

4. Compute the depth maps $z_i : \Omega_i \to \mathbb{R}^+$ with the mesh on each camera.

5. Identify the cameras where the vertex $\boldsymbol{p}_j$ is visible using the depth maps z_i.

6. Compute the per vertex uncertainty $\boldsymbol{C}_j$ using the set of visible cameras, and its projection into the normal direction σ_n (Sec. 4.5.1.4 and 4.5.1.5).

7. Compute the per pixel depth uncertainty $\hat{\sigma}_{z_i} : \Omega_i \to \mathbb{R}^+$ using geometric variations and filtering (Sec. 4.5.1.6).

Before the next section, let us just remind that our method is independent from the reconstruction process. Any geometric proxy with its associated geometric uncertainty is a valid input. Moreover, as most current reconstruction methods do not provide a geometric uncertainty, we proposed a simple method to estimate it.

4.5.2 Unstructured View Synthesis Model

The novel view is defined with a set of camera parameters matrices $\boldsymbol{K}_u$, $\boldsymbol{R}_u$, $\boldsymbol{t}_u$, and their camera matrix $\boldsymbol{P}_u = \boldsymbol{K}_u(\boldsymbol{R}_u|\boldsymbol{t}_u)$. The input views parameters are $\boldsymbol{K}_i$, $\boldsymbol{R}_i$, $\boldsymbol{t}_i$, and their camera matrix $\boldsymbol{P}_i = \boldsymbol{K}_i(\boldsymbol{R}_i|\boldsymbol{t}_i)$. From the geometrical proxy we obtained the depth map of each view i, $z_i : \Omega_i \to \mathbb{R}^+$. In the first subsection (4.5.2.1) we compute the set of backward warps τ_i for each input view i, warping a point $\boldsymbol{x}_i \in \Omega_i$ to $\boldsymbol{x} \in \Gamma$. We compute as well the associated visibility maps $m_i : \Omega_i \to \{0, 1\}$, and the inverse warps $\beta_i : \Gamma \to \Omega_i$. Once we have the warps, in the next subsections we compute the derivatives with respect to depth (4.5.2.2) and the derivatives with respect to space (4.5.2.3, 4.5.2.4, 4.5.2.5). All those terms are needed to compute the weight factors of our energy (Eq. 4.16), which depend on $|\det D\beta|$ (Eq. 4.22) and $\frac{\partial \tau_i}{\partial z}$ (Eq. 4.15).

4.5.2.1 Computing the Forward and Backward Warps

In order to establish the warp functions we use the geometric transformations introduced in Sec. 3.1. Given a point $\boldsymbol{x}_i$ on the image i, we transform it into the image u with the *reconstruction matrix* described in Sec. 3.1.7. As we have a depth value $z_i(\boldsymbol{x}_i)$ for each point, it is now more convenient to use the *reconstruction matrix* given by Eq. 3.24, defining the "standard" disparity representation $d = \frac{1}{z}$ proposed by Okutomi and Kanade (1993). Let us briefly recall it. Given a camera matrix $\boldsymbol{P}_i$ and its parameters $\boldsymbol{K}_i$, $\boldsymbol{R}_i$ and $\boldsymbol{t}_i$, their associated *reconstruction matrix* is

$$\tilde{\boldsymbol{P}}_i = \tilde{\boldsymbol{K}}_i \boldsymbol{E}_i, \quad \text{with} \quad \tilde{\boldsymbol{K}}_i = \begin{pmatrix} \boldsymbol{K}_i & \boldsymbol{0} \\ \boldsymbol{0}^t & 1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{E}_i = \begin{pmatrix} \boldsymbol{R}_i & \boldsymbol{t}_i \\ \boldsymbol{0}^t & 1 \end{pmatrix}. \tag{4.47}$$

The matrix $\boldsymbol{E}_i$ is a 3D rigid-body (Euclidean) transformation and $\tilde{\boldsymbol{K}}_i$ is the full-rank calibration matrix. When working with the reconstruction matrix, the normalization is done by dividing by the *third* element of the vector to obtain the normalized form $\boldsymbol{x}_i = (x_i, y_i, 1, (z_i(x_i, y_i))^{-1})$. The matrix $\tilde{\boldsymbol{P}}_i$ is a full-rank invertible matrix. Its inverse $\tilde{\boldsymbol{P}}_i^{-1}$ maps a point $\tilde{\boldsymbol{x}}_i = (x_i, y_i, 1, (z_i(x_i, y_i))^{-1})$ on the camera into a 3D point $\tilde{\boldsymbol{p}}_W$ in the world,

$$\tilde{\boldsymbol{p}}_W = \tilde{\boldsymbol{P}}_i^{-1} \tilde{\boldsymbol{x}}_i \tag{4.48}$$

Hence, starting with a point on camera i, we can transform it to the camera j by using both camera matrices $\tilde{\boldsymbol{P}}_i^{-1}$ and $\tilde{\boldsymbol{P}}_j$ with

$$\tilde{\boldsymbol{x}}_j \propto \tilde{\boldsymbol{P}}_j \tilde{\boldsymbol{P}}_i^{-1} \tilde{\boldsymbol{x}}_i, \quad \text{or their decomposition} \quad \tilde{\boldsymbol{x}}_j \propto \tilde{\boldsymbol{K}}_j \tilde{\boldsymbol{E}}_j \tilde{\boldsymbol{E}}_i^{-1} \tilde{\boldsymbol{K}}_i^{-1} \tilde{\boldsymbol{x}}_i. \tag{4.49}$$

In our case, the homogeneous version $\tilde{\tau}_i$ of the warp $\tau_i : \Omega_i \to \Gamma$ can be written as a 4×4 matrix $\tilde{\boldsymbol{T}}_i$, using the reconstruction matrices $\tilde{\boldsymbol{P}}_i$ and $\tilde{\boldsymbol{P}}_u$ of the cameras i and u:

$$\tilde{\boldsymbol{T}}_i = \tilde{\boldsymbol{P}}_u \tilde{\boldsymbol{P}}_i^{-1}. \tag{4.50}$$

As we are only interested in the final position of the warped point, we can discard the last row of $\tilde{\boldsymbol{T}}_i$, and write the resulting matrix using a row vector notation. Moreover, to simplify the notation, let us drop the camera index i and write

$$\boldsymbol{T} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tilde{\boldsymbol{T}} \quad \text{and} \quad \boldsymbol{T} = \begin{pmatrix} \boldsymbol{t}_1 \\ \boldsymbol{t}_2 \\ \boldsymbol{t}_3 \end{pmatrix}. \tag{4.51}$$

$\boldsymbol{T}$ is a 3×4 matrix and $\boldsymbol{t}_1, \boldsymbol{t}_2$ and $\boldsymbol{t}_3$ are 4 dimensional vectors. Then the image warp of a point $\tilde{\boldsymbol{x}}_i = (x_i, y_i, 1, (z_i(x_i, y_i))^{-1})$ is

$$\tau_i(\tilde{\boldsymbol{x}}_i) = \left(\frac{\boldsymbol{t}_1 \cdot \tilde{\boldsymbol{x}}_i}{\boldsymbol{t}_3 \cdot \tilde{\boldsymbol{x}}_i}, \frac{\boldsymbol{t}_2 \cdot \tilde{\boldsymbol{x}}_i}{\boldsymbol{t}_3 \cdot \tilde{\boldsymbol{x}}_i} \right). \tag{4.52}$$

The inverse function $\beta_i = \tau_i^{-1}$, is defined in the domain of visible points V_i. It warps points from Γ into Ω_i. Its 4×4 matrix $\tilde{\boldsymbol{B}}_i$ can be straightforwardly written by inverting $\tilde{\boldsymbol{T}}_i$:

$$\tilde{\boldsymbol{B}}_i = \tilde{\boldsymbol{P}}_i \tilde{\boldsymbol{P}}_u^{-1}. \tag{4.53}$$

Again, let us drop the camera index i to simplify the notation and write

$$\boldsymbol{B} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tilde{\boldsymbol{B}} \quad \text{and} \quad \boldsymbol{B} = \begin{pmatrix} \boldsymbol{b}_1 \\ \boldsymbol{b}_2 \\ \boldsymbol{b}_3 \end{pmatrix}, \tag{4.54}$$

where $\boldsymbol{B}$ is a 3x4 matrix and $\boldsymbol{b}_1, \boldsymbol{b}_2$ and $\boldsymbol{b}_3$ are 4 dimensional vectors. The forward warp of a point $\tilde{\boldsymbol{x}}_u = (x_u, y_u, 1, (z_u(x_u, y_u))^{-1})$ is

$$\beta_i(\tilde{\boldsymbol{x}}_u) = \left(\frac{\boldsymbol{b}_1 \cdot \tilde{\boldsymbol{x}}_u}{\boldsymbol{b}_3 \cdot \tilde{\boldsymbol{x}}_u}, \frac{\boldsymbol{b}_2 \cdot \tilde{\boldsymbol{x}}_u}{\boldsymbol{b}_3 \cdot \tilde{\boldsymbol{x}}_u} \right). \tag{4.55}$$

4.5.2.2 Derivative of the Warp with respect to Depth

We now compute the derivative of the warp τ_i with respect to the depth: $\frac{\partial \tau_i}{\partial z}$. We evaluate this derivative on a point $\tilde{\boldsymbol{x}}_i = (x_i, y_i, 1, (z_i(x_i, y_i))^{-1})$. Let us recall the notation $\boldsymbol{v}[i]$ to refer to the i'th element of the vector, and let us simplify the notation by writing z_i instead of $z_i(x_i, y_i)$, If we develop the dot product in Eq. 4.52, we can write the coordinates of the warped point as

$$\tau_i(\tilde{\boldsymbol{x}}_i) = \left(\frac{\boldsymbol{t}_1[1]x_i + \boldsymbol{t}_1[2]y_i + \boldsymbol{t}_1[3] + \frac{\boldsymbol{t}_1[4]}{z_i}}{\boldsymbol{t}_3[1]x_i + \boldsymbol{t}_3[2]y_i + \boldsymbol{t}_3[3] + \frac{\boldsymbol{t}_3[4]}{z_i}}, \frac{\boldsymbol{t}_2[1]x_i + \boldsymbol{t}_2[2]y_i + \boldsymbol{t}_2[3] + \frac{\boldsymbol{t}_2[4]}{z_i}}{\boldsymbol{t}_3[1]x_i + \boldsymbol{t}_3[2]y_i + \boldsymbol{t}_3[3] + \frac{\boldsymbol{t}_3[4]}{z_i}} \right) \quad (4.56)$$

Then its partial derivative with respect to z is

$$\frac{\partial \tau_i}{\partial z}(\tilde{\boldsymbol{x}}_i) = \left(\frac{-\boldsymbol{t}_1[4](\boldsymbol{t}_3 \cdot \tilde{\boldsymbol{x}}_i) + (\boldsymbol{t}_1 \cdot \tilde{\boldsymbol{x}}_i)\boldsymbol{t}_3[4]}{z_i^2(\boldsymbol{t}_3 \cdot \tilde{\boldsymbol{x}}_i)^2}, \frac{-\boldsymbol{t}_2[4](\boldsymbol{t}_3 \cdot \tilde{\boldsymbol{x}}_i) + (\boldsymbol{t}_2 \cdot \tilde{\boldsymbol{x}}_i)\boldsymbol{t}_3[4]}{z_i^2(\boldsymbol{t}_3 \cdot \tilde{\boldsymbol{x}}_i)^2} \right) \quad (4.57)$$

4.5.2.3 Jacobian of the Warp

We now write the derivatives with respect to the image space $\frac{\partial \tau_i}{\partial x}$ and $\frac{\partial \tau_i}{\partial y}$. Let us recall that the depth of a point on the image is given by the function $z_i : \Omega_i \to \mathbb{R}$. The terms $\frac{\partial z_i}{\partial x}$ and $\frac{\partial z_i}{\partial y}$ will appear in the computations as a consequence of the chain rule. With an abuse of notation we also write $z_i(\tilde{\boldsymbol{x}}_i)$, $\frac{\partial z_i}{\partial x}(\tilde{\boldsymbol{x}}_i)$ and $\frac{\partial z_i}{\partial y}(\tilde{\boldsymbol{x}}_i)$. We make this dependence explicit by writing

$$\tau_i(\tilde{\boldsymbol{x}}_i) = \left(\frac{\boldsymbol{t}_1[1]x_i + \boldsymbol{t}_1[2]y_i + \boldsymbol{t}_1[3] + \frac{\boldsymbol{t}_1[4]}{z_i(\tilde{\boldsymbol{x}}_i)}}{\boldsymbol{t}_3[1]x_i + \boldsymbol{t}_3[2]y_i + \boldsymbol{t}_3[3] + \frac{\boldsymbol{t}_3[4]}{z_i(\tilde{\boldsymbol{x}}_i)}}, \frac{\boldsymbol{t}_2[1]x_i + \boldsymbol{t}_2[2]y_i + \boldsymbol{t}_2[3] + \frac{\boldsymbol{t}_2[4]}{z_i(\tilde{\boldsymbol{x}}_i)}}{\boldsymbol{t}_3[1]x_i + \boldsymbol{t}_3[2]y_i + \boldsymbol{t}_3[3] + \frac{\boldsymbol{t}_3[4]}{z_i(\tilde{\boldsymbol{x}}_i)}} \right). \quad (4.58)$$

The components $(k = 1, k = 2)$ of the partial derivatives with respect to x are

$$\frac{\partial \tau_i}{\partial x}(\tilde{\boldsymbol{x}}_i)[k] = \frac{\left(\boldsymbol{t}_k[1] - \frac{\boldsymbol{t}_k[4]}{z_i^2(\tilde{\boldsymbol{x}}_i)} \frac{\partial z_i}{\partial x}(\tilde{\boldsymbol{x}}_i)\right) \boldsymbol{t}_3 \cdot \tilde{\boldsymbol{x}}_i - \boldsymbol{t}_k \cdot \tilde{\boldsymbol{x}}_i \left(\boldsymbol{t}_3[1] - \frac{\boldsymbol{t}_3[4]}{z_i^2(\tilde{\boldsymbol{x}}_i)} \frac{\partial z_i}{\partial x}(\tilde{\boldsymbol{x}}_i)\right)}{(\boldsymbol{t}_3 \cdot \tilde{\boldsymbol{x}}_i)^2}. \quad (4.59)$$

The components $(k = 1, k = 2)$ of the partial derivatives with respect to y are

$$\frac{\partial \tau_i}{\partial y}(\tilde{\boldsymbol{x}}_i)[k] = \frac{\left(\boldsymbol{t}_k[2] - \frac{\boldsymbol{t}_k[4]}{z_i^2(\tilde{\boldsymbol{x}}_i)} \frac{\partial z_i}{\partial y}(\tilde{\boldsymbol{x}}_i)\right) \boldsymbol{t}_3 \cdot \tilde{\boldsymbol{x}}_i - \boldsymbol{t}_k \cdot \tilde{\boldsymbol{x}}_i \left(\boldsymbol{t}_3[2] - \frac{\boldsymbol{t}_3[4]}{z_i^2(\tilde{\boldsymbol{x}}_i)} \frac{\partial z_i}{\partial y}(\tilde{\boldsymbol{x}}_i)\right)}{(\boldsymbol{t}_3 \cdot \tilde{\boldsymbol{x}}_i)^2} \quad (4.60)$$

With Eqs. 4.59 and 4.60 we obtain the 2×2 Jacobian matrix. As the inverse transfer function β_i can be written by using the inverse matrix $\tilde{\boldsymbol{T}}_i^{-1} = \tilde{\boldsymbol{B}}_i$, all the previous computations can be directly done on the forward warp β_i. One just needs to replace τ_i with β_i, $\boldsymbol{t}$ with $\boldsymbol{b}$, $\tilde{\boldsymbol{x}}_i$ with $\tilde{\boldsymbol{x}}_u$ and $z_i : \Omega_i \to \mathbb{R}^+$ with $z_u : \Gamma \to \mathbb{R}^+$ in Eq. 4.59 and 4.60. From the Jacobian matrix, the expressions $|\det D\beta_i|$ and $|\det D\beta_i|''$ (Eq. 4.25) can be computed.

Finite differences on image domain vs. approximated mesh normal The deformation weight $|\det D\beta_i|$ relies on the computation of the partial derivatives $\frac{\partial z_u}{\partial x}$ and $\frac{\partial z_u}{\partial y}$. These can either be computed by finite differences in the image domain

Γ, or directly from the normal vector to the surface. In general both methods yield similar numerical results. However, important differences may appear at the disocclusion borders. In this case the discrete image difference compares two depth values from a foreground and a background location, thus yielding an approximation of the geometry with a very slanted plane with respect to the view, e.g. almost parallel to the viewing rays. As the partial derivatives $\frac{\partial z_u}{\partial x}$ and $\frac{\partial z_u}{\partial y}$ values may be very high, it is common in the implementation to threshold the computed values with an arbitrarily chosen maximum. This effect does not appear when we compute with the interpolated 3D normals, because the interpolation is performed with nearby vertices. Moreover, the mesh resolution may be higher than the image resolution, thus providing better estimate of the warp deformation.

In the case where the geometric proxy is given as a 3D model, and not as a set of depth maps, the use of surface normal vectors has another computational advantage. To compute the depth maps we need a first render pass, and then a second pass to compute the finite differences. Locally approximating the surface with the normal vectors allows us to compute the derivatives in a single pass.

4.5.2.4 Depth of an Image Point with the Tangent Plane to the Geometry Surface and its Derivatives

Let us consider the tangent plane to the geometric proxy surface at the world point $\boldsymbol{p}_W$ given by the normal vector $\boldsymbol{n}_W$, and a camera with parameters $\boldsymbol{K}, \boldsymbol{R}$ and $\boldsymbol{t}$. Let us compute the depth map $z(\boldsymbol{x})$ at a generic image point $\boldsymbol{x} = (x, y)$ defined by this tangent plane, as well as its spatial derivatives: $\frac{\partial z}{\partial x}$ and $\frac{\partial z}{\partial y}$.

First we move into the camera frame, where the camera is centered at the origin and looking into the positive z axis. In this frame the 3D point $\boldsymbol{p}_C$ and the transformed normal $\boldsymbol{n}_C$ are

$$\boldsymbol{p}_C = \boldsymbol{R}\, \boldsymbol{p}_W + \boldsymbol{t} \quad \text{and} \quad \boldsymbol{n}_C = \boldsymbol{R}\, \boldsymbol{n}_W. \tag{4.61}$$

Points $\boldsymbol{p}$ on the plane fulfill

$$(\boldsymbol{p} - \boldsymbol{p}_C) \cdot \boldsymbol{n}_C = 0. \tag{4.62}$$

The camera viewing ray through an image point (x, y) can be written in parametric form as

$$\boldsymbol{p}_\lambda = \lambda\, \boldsymbol{K}^{-1} \bar{\boldsymbol{x}}, \tag{4.63}$$

with $\lambda \in \mathbb{R}^+$. The intersection of the plane with the viewing ray is obtained by substituting Eq. 4.63 in Eq. 4.62 and solving for λ. We obtain

$$\lambda_p(\boldsymbol{x}) = \frac{\boldsymbol{n}_C \cdot \boldsymbol{p}_C}{\boldsymbol{n}_C \cdot\ \boldsymbol{K}^{-1} \bar{\boldsymbol{x}}} \quad \text{and} \quad \boldsymbol{p}_p(\boldsymbol{x}) = \lambda_p(\boldsymbol{x})\, \boldsymbol{K}^{-1} \bar{\boldsymbol{x}}. \tag{4.64}$$

The depth map $z(\boldsymbol{x})$ is given by the third coordinate of $\boldsymbol{p}_p$.

Let us now introduce $\boldsymbol{k}_1, \boldsymbol{k}_2$ and $\boldsymbol{k}_3$, denoting the three columns of the matrix

$\boldsymbol{K}^{-1}$:

$$\boldsymbol{K}^{-1} = \begin{pmatrix} \boldsymbol{k}_1 & \boldsymbol{k}_2 & \boldsymbol{k}_3 \end{pmatrix} \tag{4.65}$$

The partial derivative of $\boldsymbol{p}_p$ with respect to x is

$$\frac{\partial \boldsymbol{p}_p}{\partial x}(\boldsymbol{x}) = \frac{\partial \lambda_p}{\partial x}(\boldsymbol{x})\boldsymbol{K}^{-1}\bar{\boldsymbol{x}} + \lambda_p(\boldsymbol{x})\boldsymbol{k}_1, \tag{4.66}$$

where

$$\frac{\partial \lambda_p}{\partial x}(\boldsymbol{x}) = \frac{(-\boldsymbol{n}_C \cdot \boldsymbol{p}_C)(\boldsymbol{n}_C \cdot \boldsymbol{k}_1)}{(\boldsymbol{n}_C \cdot (\boldsymbol{K}^{-1}\bar{\boldsymbol{x}}))^2}. \tag{4.67}$$

The partial derivative of $\boldsymbol{p}_p$ with respect to y is

$$\frac{\partial \boldsymbol{p}_p}{\partial y}(\boldsymbol{x}) = \frac{\partial \lambda_p}{\partial y}(\boldsymbol{x})\boldsymbol{K}^{-1}\bar{\boldsymbol{x}} + \lambda_p(\boldsymbol{x})\boldsymbol{k}_2, \tag{4.68}$$

where

$$\frac{\partial \lambda_p}{\partial y}(\boldsymbol{x}) = \frac{(-\boldsymbol{n}_C \cdot \boldsymbol{p}_C)(\boldsymbol{n}_C \cdot \boldsymbol{k}_2)}{(\boldsymbol{n}_C \cdot (\boldsymbol{K}^{-1}\bar{\boldsymbol{x}}))^2}. \tag{4.69}$$

The Eq. 4.59 and 4.60 can now be computed by setting $\frac{\partial z_i}{\partial x}(\tilde{\boldsymbol{x}}) = \frac{\partial \boldsymbol{p}_p}{\partial x}(x_i, y_i)[3]$, $\frac{\partial z_i}{\partial y}(\tilde{\boldsymbol{x}}) = \frac{\partial \boldsymbol{p}_p}{\partial y}(x_i, y_i)[3]$ and substituting the generic parameters $\boldsymbol{K}, \boldsymbol{R}, \boldsymbol{t}$ with the ones used in the τ_i computation: $\boldsymbol{K}_i, \boldsymbol{R}_i$ and $\boldsymbol{t}_i$.

4.5.2.5 Integrating the Optics Distortion in the Transfer Function

The input cameras of our algorithm are generic. For the sake of simplicity, we assumed that they follow a pinhole camera model, which supposes that there is no optical distortion in the image formation process. Distorted images may come from fish-eye or panoramic cameras, which cannot be properly represented by a pinhole camera. Most of the Image Based Rendering methods (Shum *et al.*, 2007; Kopf *et al.*, 2014) assume that the images are first undistorted as a pre-processing step, usually during calibration. However, the correction of the optical distortion may add some blurriness in the corrected images, as some pixels, specially near the borders of the image, can be (strongly) stretched. Our method considers the warp function τ from one image to the other, so, it is possible to integrate the optical distortion correction into the warp function. This allows us to work with the raw *distorted* images of the camera, instead of their undistorted version.

Let us first introduce a generic radial distortion model, and then analyze the impact of the undistortion warp into the weights from Eq. 4.26.

We note a generic radial distortion warp from an undistorted pixel $\boldsymbol{x}^u$ into a distorted one $\boldsymbol{x}^d$ as $\mathcal{D}$. Its form is

$$\mathcal{D}(\boldsymbol{x}^u) = \boldsymbol{e} + \lambda(||\boldsymbol{x} - \boldsymbol{e}||)(\boldsymbol{x} - \boldsymbol{e}), \tag{4.70}$$

where $\boldsymbol{x}^u$ is the undistorted pixel, $\boldsymbol{e}$ the center of distortion and $\lambda : \mathbb{R}^+ -> \mathbb{R}^+$ the distortion ratio. λ is in general assumed to be monotonic (Hartley and Kang, 2007), allowing to define the inverse warp $\mathcal{U}(\boldsymbol{x}^d) = \boldsymbol{x}^u$. The undistortion warp transforms a distorted pixel $\boldsymbol{x}^d$ into an undistorted one $\boldsymbol{x}^u$.

Now that we have characterized the optical distortion and undistortion warps, let us integrate them into our energy equations (Eq. 4.26). A generic warp τ_i^d taking into account the optical distortion can be written as the composition of the undistortion warp $\mathcal{U}_i$, the pinhole camera warp τ_i, and the distortion warp $\mathcal{D}_u$ of the rendered image u:

$$\tau_i^d(\boldsymbol{x}^d) = (\mathcal{D}_u \circ \tau_i \circ \mathcal{U}_i)(\boldsymbol{x}^d). \tag{4.71}$$

In general, we do not want to render an image u with optical distortion. To assume no distortion in the u image does not substantially change the following equations, and of course, one could chose to generate images with distortion. Equations get just less readable because of the double function composition, so we consider $\mathcal{D}_u$ to be the identity warp and

$$\tau_i^d(\boldsymbol{x}^d) = (\tau_i \circ \mathcal{U}_i)(\boldsymbol{x}^d). \tag{4.72}$$

The forward warp map β_i^d is then

$$\beta^d(\boldsymbol{x}_u) = (\mathcal{D}_i \circ \beta_i)(\boldsymbol{x}^{\boldsymbol{u}}). \tag{4.73}$$

The weight ω_i in Eq. 4.18 depends on the derivative of τ_i^d with respect to z. The derivative of τ_i^d with respect to z at $\boldsymbol{x}^d$ is equal to the derivative of τ with respect to z at $\boldsymbol{x}^u$, because $\mathcal{U}_i$ does not depend on the depth of the point:

$$\frac{\partial \tau_i^d}{\partial z}(\boldsymbol{x}^d) = (\frac{\partial \tau_i}{\partial z} \circ \mathcal{U}_i)(\boldsymbol{x}^d). \tag{4.74}$$

This can easily be seen by rewriting Eq. 4.56 with $\mathcal{U}_i(x_i^d, y_i^d)$ instead of x_i, y_i and deriving Eq. 4.57.

We are also interested in the Jacobian of the forward warp map β_i^d. The Jacobian of a composition $(\mathcal{D}_i \circ \beta_i)$ is given by the product of their Jacobians, (evaluated at the proper points)

$$D(\beta_i^d)(\boldsymbol{x}_u) = D\mathcal{D}_i(\beta_i(\boldsymbol{x}_u))\ D\beta_i(\boldsymbol{x}_u). \tag{4.75}$$

In addition, the determinant of a matrix product is given by the product of determinant of each matrix, so

$$|\det D\beta_i^d| = |\det D\mathcal{D}_i||\det D\beta_i|. \tag{4.76}$$

The integration of the optical distortion in our generic warps is quite simple.

4.5.2.6 Closure on Transfer Functions and Partial Derivatives

With the obtained image warps and their derivatives we can compute all the necessary terms of our energy Eq. 4.16, including the weight factors $|\det D\beta|$ and

$\omega_i(\boldsymbol{x})$.

4.5.3 Datasets

We numerically evaluate the unstructured and generic version of our method with scenes from three different datasets. In a first set of experiments we use two scenes from a *dense multi-view stereo dataset* (Strecha *et al.*, 2008): *fountain-P11* and *castle-P19*. Their dataset provides a set of unstructured images, together with their calibration matrices P_i. The *fountain-P11* dataset contains 11 images, and the *castle-P19* dataset 19. We also created the dataset *Hercules* with images taken in the "Chateau de Vizille" gardens, in France. This dataset consists of 52 images. In Fig. 4.14 we show several images of the datasets.

For each dataset we consider different qualities of geometric proxy. We first consider the best reconstruction available for the dataset, that we label G0. For the dataset *fountain-P11* we use the 3D reconstruction created from laser scans, which are unfortunately not publicly available for the dataset *castle-P19*. For the dataset *Hercules* we use all 52 images to create the 3D reconstruction with CMVS (Furukawa *et al.*, 2010). Although this is our "best" reconstruction, we should avoid to address it as "ground truth", as the reconstruction created with laser scans may contain holes in the geometry, and the 3D reconstruction obtained with CMVS does not represent a "ground truth". In addition to G0, we create 3 different geometric proxys with "less reliable" qualities. For *Hercules* we reduce the number of images in the reconstruction by half, i.e. 26. For each dataset we create the geometry G1 with the full resolution images, the geometry G2 with the images downsampled with a $\times 2$ factor, and the geometry G3 with the images downsampled with a $\times 4$ factor. For each geometry of each dataset, we compute the per pixel uncertainty as described in Sec. 4.5.1.4, 4.5.1.5 and 4.5.1.6.

4.5.4 Numerical Evaluation

To compare the results obtained with our method, we implemented the *Unstructured Lumigraph Rendering* (ULR) (Buehler *et al.*, 2001), as well as the generalization of the method proposed by Wanner and Goldluecke (2012) (SAVS) into the unstructured camera configuration. The parameters for URL were chosen as described in the original paper: $K = 4$ and $\beta = 0.05$. The parameter λ leveraging the data term and the prior in the energy from Eq. 4.16 are set to 0.05. All parameters are kept constant for all datasets.

To evaluate the methods we render a view from the dataset, without using it as an input for the algorithm.[1] We measure the PSNR and DSSIM values between the actual and the generated images. In the unstructured configuration it is common that some parts of the rendered image are not seen by any other camera in the dataset. In our computation these parts of the image are "inpainted" by the TV prior (Sec. 4.3.2.4). To avoid evaluating the methods in these areas, we use a

[1] For these experiments we could not render images at a higher resolution due a problem in the implementation of the code. We hope to solve this problem in the near future.

Fig. 4.14: *Images from the 3 dataset used for evaluation. First row: fountain-P11 dataset. Second row: castle-P19 dataset. Third and fourth row: Hercules dataset.*

visibility mask to only evaluate PSNR and DSSIM values on pixels which are at least visible in one image of the dataset.

4.5.5 Processing Time

The resolution of the rendered views are 1536×1024 for the *fountain-P11* and the *castle-P19* datasets, and 2376×1584 for the *Hercules* dataset. The computation time is decomposed in three steps. The input generation (camera calibration, reconstruction and per pixel uncertainty computation) is done offline and may take in the order of 1 to 2 hours. The computation of the warps τ_i and the magnitudes for the weights is performed in real time. Those magnitudes are $|\det D\beta_i|$, $|\det D\beta_i|''$, $\frac{\partial \tau_i}{\partial z}$, as well as the angular deviation and resolution penalties of the ULR. The energy minimization step is on the order of 10 to 12 seconds for our method and 1 to 2 seconds for SAVS, for an input of 10 images at 1536×1024 resolution. All experiments used an nVidia GTX Titan GPU.

4.5.6 Generic Configuration Results

In Table 4.2 we show the numerical results obtained with the three methods. For the large majority of rendered images the best performing algorithm is either ULR or the proposed one. This result is coherent with the fact that those algorithms do consider the *angular deviation* in their equations. However, as with previous experiments, numerical results should be interpreted carefully, as the difference in PSNR and DSSIM between the different methods is relatively small.

In Fig. 4.15 we show detailed closeups illustrating the benefits of the inclusion of the *angular deviation* in the method. The generalization of the method proposed by Wanner and Goldluecke (2012) produces noticeable artifacts, which become more visible when the geometric proxy becomes less accurate (G3). As the *angular deviation* is not taken into account, all images are blended together. Moreover, images with an important angular deviation may have a higher weight because of the foreshortening effects accounted by $|\det D\beta_i|$.

The ULR method performs globally at best. As only a few images ($K = 4$) are considered in the final blend, the generated images are sharper. The small number of views in the final blend allows to avoid most of the artifacts arising on the SAVS or our Proposed method. However, as illustrated in Fig. 4.16, to only blend a low number of views can also create important artifacts, specially if the geometric proxy is not accurate (G3).

Our method provides sharp images similar to those obtained by ULR. The generated images still include artifacts at the same locations as SAVS. Because the *angular deviation* is taken into account, those artifacts are reduced with respect to SAVS but not completely removed. Even if the contribution of a camera is close to zero, if the proposed color is very different from the other cameras, the obtained blend might be wrong (see Fig. 4.17).

Although the results provided by our method are globally close to ULR, some artifacts appear at certain locations (see Fig. 4.18). Those artifacts arise because the

term σ_{g_i} computed with the ∇u can rapidly change from one pixel to its neighbor. Thus the balance between the *angular deviation* and the *resolution* is not continuous in the image domain and might abruptly change from one pixel to another. This artifacts are specially noticeable when the geometric proxy is not accurate and the proposed colors by the input images are very different.

4.5.7 Discussion and Hints for Improvement

The benefits of the proposed method with respect to SAVS are evident from the obtained results. Generated images are sharper and contain less artifacts.

The benefits of the proposed method with respect to ULR are, in terms of the generated images, less evident. Surprisingly, the adaptability of our method to the precision of the geometric uncertainty did not translate into a better final image as it did in the structured configuration. Further investigation is needed to understand if the proposed uncertainty estimation was not accurate enough or if the problem lies in the model itself.

An important advantage of our method with respect to ULR is to avoid the parameter K defining how many cameras should be used in the final blend. As pointed out by other techniques (Davis *et al.*, 2012; Kopf *et al.*, 2014), when generating a sequence of images by moving the virtual camera, strong transition artifacts arise in ULR when switching from one set of cameras to another. In our method, those transitions are smooth as the complete set of images is considered.

Let us summarize the three main issues unsolved by our technique. The dependency on the latent image u implies two main drawbacks. The first is the appearance of artifacts illustrated in Fig. 4.18. The second is that we need an iterative reweighted method in order to minimize the energy, which is considerably longer compared to the minimization of Wanner and Goldluecke (2012), and pretty much longer compared to the direct blend performed in Buehler *et al.* (2001). In order to address these issues we propose directions for future work.

Dependency on the latent image The dependency on the latent image u was very helpful in the Lightfield configuration, allowing to better render color edges. However, in the general configuration, the local computation of ∇u generated artifacts where input colors did not agree (see Fig. 4.18). In order to minimize those artifacts one could try to smooth the computation of ∇u. Neighboring pixels would have more similar values and different camera contributions would become more homogeneous. Although this smoothing step could improve the results, it would imply the addition of a parameter in our model, thus breaking our effort to achieve a parameter-free method.

Another way to avoid the dependency on the latent image would be to only consider the length of $\frac{\partial \tau_i}{\partial z}$ corresponding to the length of the projected epipolar line and disregard the color variation. The benefit of color edges would be lost, but the weights of neighboring pixels would be more consistent. Again, although this simplification could improve the results we do not have yet any physical evidence to sustain such a choice.

	fountain-P10						castle-P19				Hercules					
	img04		*img05*		*img06*		*img06*		*img08*		*img00*		*img03*		*img11*	
G0																
SAVS	32.2	50	33.17	48	29.32	60	N/A	N/A	N/A	N/A	14.98	**2064**	15.79	2185	14.43	1993
ULR	**33.7**	**37**	**34**	**35**	**32.7**	**39**	N/A	N/A	N/A	N/A	**15.28**	2068	**16.04**	2196	14.6	2012
Proposed	32.85	44	33.86	42	29.71	53	N/A	N/A	N/A	N/A	15.26	2077	16.03	**2178**	**14.85**	**1984**
G1																
SAVS	27.7	59	**27.5**	60	25.05	86	23.68	240	21.83	226	14.48	2129	15.32	2327	14.07	2110
ULR	**27.84**	**55**	27.32	**57**	**25.69**	**72**	**26.79**	**181**	22.78	**175**	14.55	2167	15.39	2376	13.98	2156
Proposed	27.39	59	27.15	61	24.79	86	24.81	227	**22.9**	179	**14.77**	**2128**	**15.49**	**2325**	**14.46**	**2084**
G2																
SAVS	28.18	65	**26.98**	73	24.95	96	23.26	249	21.47	236	14.63	2116	15.43	2302	13.9	1881
ULR	**28.59**	**53**	26.91	**64**	**25.84**	**76**	**26.25**	**190**	22.62	**174**	14.82	2130	15.55	2332	13.74	1913
Proposed	28.27	59	26.88	68	25.15	88	24.92	224	**22.8**	176	**15.02**	**2087**	**15.75**	**2267**	**14.21**	**1853**
G3																
SAVS	27.41	78	26.41	82	24.37	119	22.65	290	21.07	275	14.6	2144	15.39	2300	13.61	**1899**
ULR	**28.22**	**60**	**27.35**	**64**	**27.01**	**68**	**25.68**	**217**	22.47	195	14.82	2158	15.63	2316	13.31	1931
Proposed	28.15	64	27.06	68	26.02	81	24.47	253	**22.69**	**193**	**15.01**	**2118**	**15.75**	**2268**	**13.7**	1903

Table 4.2: *Numerical results for novel view synthesis with an unstructured configuration of cameras. We compare our method to Wanner and Goldluecke (2012) (SAVS) and Buehler* et al. *(2001) (ULR). G0, G1, G2 and G3 correspond to different qualities of the geometrical proxy. For each rendered image, the first value is the PSNR (bigger is better), the second value is DSSIM in units of* 10^{-4} *(smaller is better). The best value is highlighted in bold. See text for a detailed description of the experiments.*

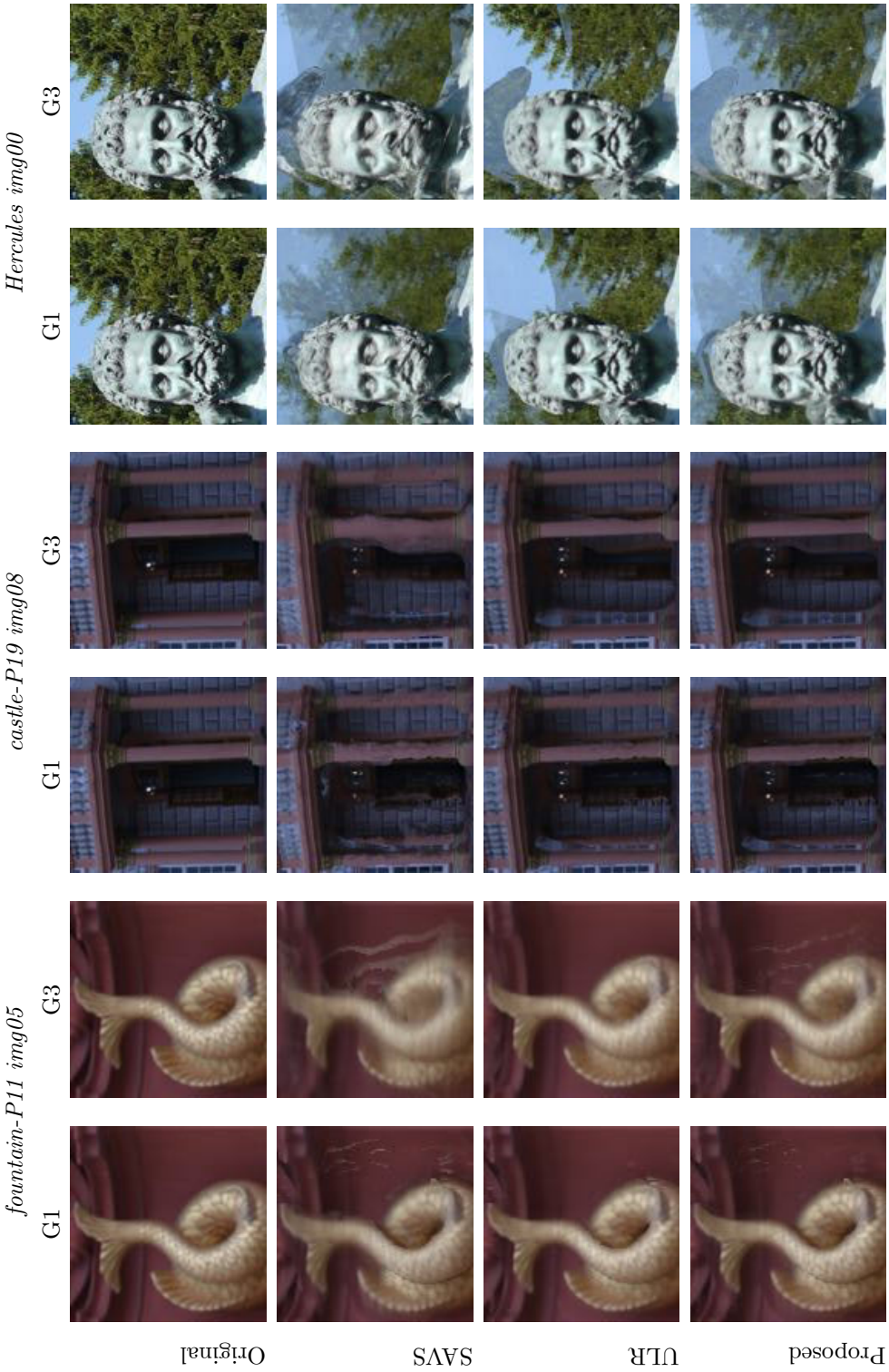
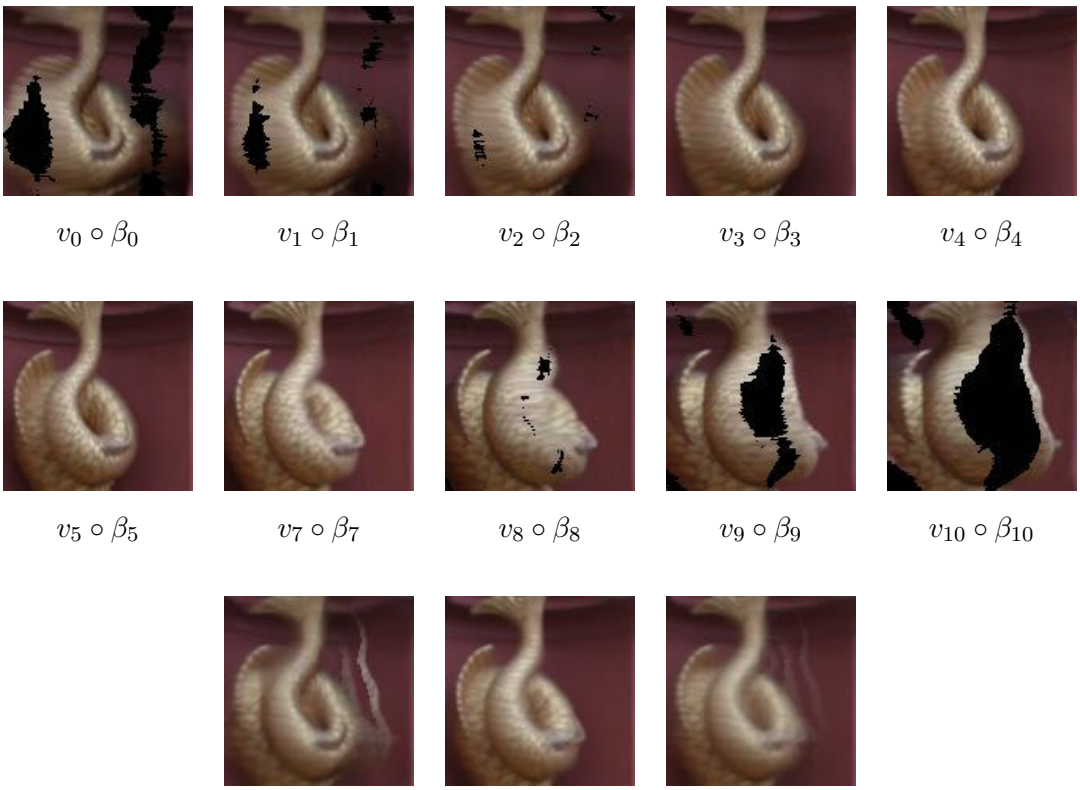


Fig. 4.15: *Visual comparison of the novel views obtained for different datasets. From top to bottom, the rows present closeups of the Original view, the results obtained by Wanner and Goldluecke (2012) (SAVS), the results obtained by Buehler et al. (2001) (ULR), and our results (Proposed). G1, G3 stand for different geometric reconstructions described in Sec. 4.5.3. Full resolution images can be found in Appendix C.*

Original SAVS ULR Proposed

Fig. 4.16: *Detail of a generated image 05 of the fountain-P11 dataset with the geometry G3. Because of the poorly geometric reconstruction, no method is capable to render the right corner of the fountain at the correct location. ULR uses only a few number of views and the corner of the fountain appears in the background. In the images generated by SAVS and our method, as more input images are blended together, the corner in the background fades away.*



$v_0 \circ \beta_0$ $v_1 \circ \beta_1$ $v_2 \circ \beta_2$ $v_3 \circ \beta_3$ $v_4 \circ \beta_4$

$v_5 \circ \beta_5$ $v_7 \circ \beta_7$ $v_8 \circ \beta_8$ $v_9 \circ \beta_9$ $v_{10} \circ \beta_{10}$

SAVS ULR Proposed

Fig. 4.17: *Blending incorrect colors. First and second rows: closeup of the input images warped into the target image with the geometry G3: $v_i \circ \beta_i$. Third row: closeup of the obtained results with the three methods. Incorrect colors are proposed for blend by input images 9 and 10. ULR only uses views 4,5 and 7 and avoids major visual artifacts. SAVS and Proposed use all the views. Visible artifacts are introduced by the views 9 and 10.*

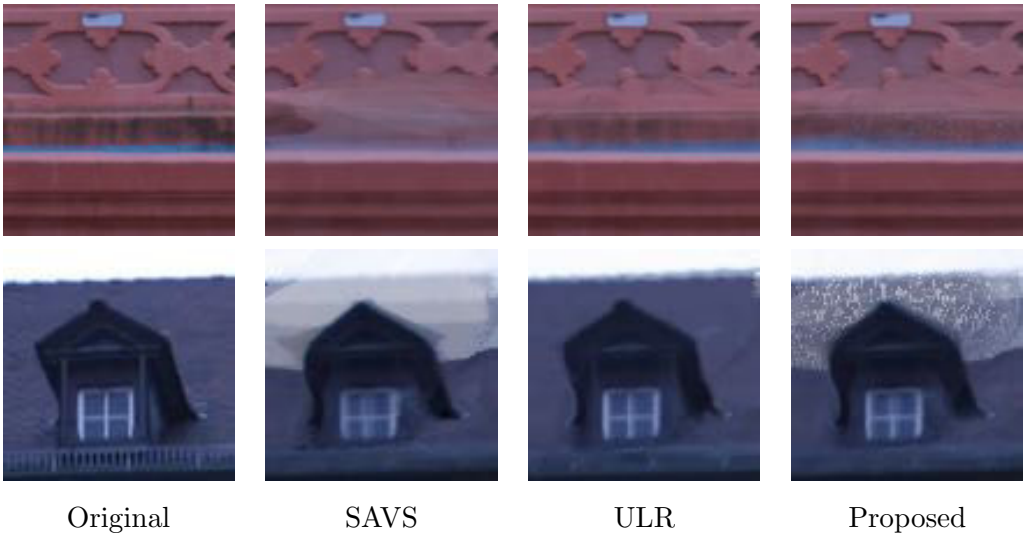


Fig. 4.18: *First row: detail of a generated image 05 of the fountain-P11 dataset with the geometry G3. Second row: detail of a generated image 06 of the castle-P19 dataset with the geometry G3. The geometry is poorly reconstructed and the proposed colors by the input images are very different. Artifacts appear in the generated images with our proposed method. The balance between the angular deviation and the resolution abruptly changes from one pixel to another, due to the value of* ∇u*. The final color can also be very different between neighboring pixels.*

Direct vs. Variational Framework In our method the weights are deduced using the Bayesian formalism. However, if super-resolution is not important, they could be used in a direct framework as proposed by Buehler *et al.* (2001). In our exploratory work (Pujades and Devernay, 2014), we tested the impact of the weights and the method when rendering a new image in the case of viewpoint interpolation. We observed that for the two camera case, blending the images with very different weights leads to very similar results. From the results in Sec. 4.5.6, it seems obvious that the weighting factor should yet have a strong relevance on the quality of the rendered image when using more than two input images. However it is unclear if the variational framework would provide better results than the direct framework. Future work should explore how the obtained results for viewpoint interpolation extend to the multiple camera configuration. If a direct blend provides equivalent results to the variational framework, computation times could be reduced to nearly real-time.

Dealing with outliers An important issue of the proposed model is that outliers are not taken into account. In our method we do not include an artifact removal process as it is common in the DIBR literature (Zinger *et al.*, 2010). As we illustrate in Fig. 4.17, artifacts arise due to incorrect colors proposed by the input views. Note that this issue is common to all presented approaches (Wanner and Goldluecke, 2012; Buehler *et al.*, 2001). Results obtained by Buehler *et al.* (2001) have less artifacts because of the small value of the parameter K.

In general, when solving a least squares problem, an outlier strongly modifies the

final estimate. However, the scientific community has developed approaches to deal with outliers (Fischler and Bolles, 1981). For example, in the direct blend of Buehler *et al.* (2001), rather than to consider the weighted mean of the input colors, one could use a mean-shift clustering technique (Cheng, 1995) to extract more robust candidates, as proposed by Fitzgibbon *et al.* (2005). Another possibility to compute the color of the final pixel could be to select one color proposed by an image, rather than to blend the input colors. The blending problem would then be transformed into a labeling problem, where each pixel of the final image would be associated with the index of the input image (Agarwala *et al.*, 2004). The proposed energy could still be used in such a framework.

Better generative models Although all those possibilities could improve the results, it remains unclear how they could be justified in a formal way. The research of better generative models must continue. An obvious lead, is to drop the Lambertian assumption. Extending the model to non-Lambertian scenes is crucial but quite hard. One would need to include general BRDF and lighting information to correctly model the transformation between input and novel views.

Each one of these leads could be a direction to be pursued in future work.

4.6 Relation to the Principles of IBR

Now that we have presented our method and evaluated its performance, let us carefully establish the links of the proposed energy with the "*desirable properties*" of IBR stated in Buehler *et al.* (2001).

As we see in Eq. 4.26, the weighting factor for each view is composed of two terms. The term $|\det D\beta_i|$ is the same as in Wanner and Goldluecke (2012) and corresponds to a measure of image deformation: it is the area of a pixel from u projected to v_i. We can formulate the intuition behind it as *how much does the observed scene change when the viewpoint changes?*

The term $\omega_i(u)$ corresponds to the depth uncertainty, as was explained in Sec. 4.3.2.3. The intuition behind this is: *how much does the observed scene change if the measured depth changes?*

4.6.1 Use of Geometric Proxies & Unstructured Input

The geometric proxies are incorporated via the warp maps τ_i, and the input can be unstructured (i.e. a random set of views in generic position). Let us recall, that although the blur kernel b and the sensor noise σ_s where considered identical for all cameras to simplify notation (Sec. 4.3.2.2), they are fully general. All cameras may have different resolutions and different sensor noise. Moreover, we do not only take into account the geometric proxies, but their associated uncertainty. This fact allows our method to cover most of the "IBR Continuum" (see Sec. 4.2.1). Our

method can take advantage of a very precise geometry from laser scans or depth sensors, and adapt to a very coarse geometric approximation (see Sec. 4.4.2).

4.6.2 Epipole Consistency

Epipole Consistency is satisfied. As explained in Sec. 4.3.2.3, the weighting factor $\omega_i(u)$ is maximal as soon as the optical rays from x_i and x are identical, so that if a camera has its epipole at x, then the contribution of this camera at x via the $\omega_i(u)$ term is higher. Although Buehler *et al.* (2001) claim that *the ideal algorithm should return a ray from the source image*, if one takes into account the sensor noise, the sampled ray from the source images could not be perfect. In our opinion, if more rays can help in the ray reconstruction they should be used, specially if resolution sensitivity varies between the contributing cameras. Of course if one considers the sensor noise to be strictly zero, then the contribution of an optical ray with epipole consistency is infinity.

4.6.3 Minimal Angular Deviation

This heuristic is provided by σ_{g_i} from Eq. 4.15. If all other dimensions are kept constant (resolution, distance to the scene, etc.), then the magnitude of the vector $\partial\tau_i/\partial z_i$ in Eq. 4.15 is exactly proportional to the sine of angle α_i between the optical rays from both cameras to the same scene point.

Fig. 4.19 illustrates two cameras at c_1 and c_2 with the same focal distance f. We choose their focal plane to be parallel to the segment between c_1 and c_2 to enforce both cameras to have the same resolution. Their distance z to the scene is thus the same. The geometric uncertainties from each camera are also chosen to be equal, $\sigma_1 = \sigma_2$. With this configuration the angular deviation α_i is a function of the optical center distance $c - c_i$ and the depth z of the observed element $\tan(\alpha_i) = \frac{c-c_i}{z}$. The

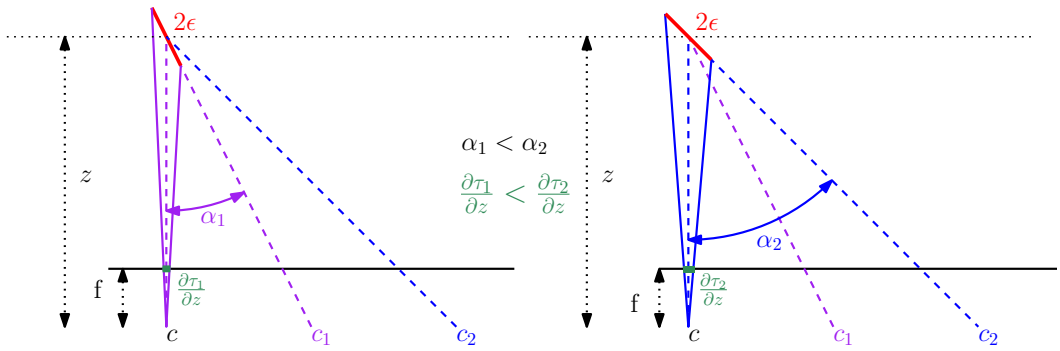


Fig. 4.19: *Two cameras, at c_1, c_2 with focal length f, observing the same point at depth z. In order to analyze the impact of the angle into the final weight we disregard the foreshortening effects. The only difference between the views is the angle of observation α_1 and α_2. The resulting magnitude of $\frac{\partial\tau_i}{\partial z}$ is smaller if the angle is smaller. It is exactly proportional to the sine of the angle α_i.*

derivative of τ_i with respect to z can be computed as the limit

$$\lim_{\varepsilon \to 0} \frac{\tau_i(z+\varepsilon) - \tau_i(z-\varepsilon)}{2\varepsilon}, \tag{4.77}$$

or as a function of the angle α_i

$$\lim_{\varepsilon \to 0} f\left(\frac{z\sin(\alpha_i)}{z^2 - \varepsilon^2\cos^2(\alpha_i)}\right) = \sin(\alpha_i)\frac{f}{z}. \tag{4.78}$$

The weight ω_i as a function of α_i is then

$$\omega_i = \frac{1}{{\sigma_s}^2 + \left(\sigma_{z_i}\sin(\alpha_i)\frac{f}{z}\right)^2}. \tag{4.79}$$

The ratio between the contributions of two cameras, i and j, is then

$$\frac{\omega_i}{\omega_j} = \frac{{\sigma_s}^2 + \left(\sigma_{z_j}\sin(\alpha_j)\frac{f}{z}\right)^2}{{\sigma_s}^2 + \left(\sigma_{z_i}\sin(\alpha_i)\frac{f}{z}\right)^2}. \tag{4.80}$$

Buehler *et al.* (2001) state that *source image rays with similar angles to the desired ray should be used when possible.* Their proposed *angular deviation* penalty linearly depends on the angle α_i, whereas in our case, the penalty depends on the square sinus of the angle. The proposed weights stronger penalize distant angles.

4.6.4 Resolution Sensitivity

This heuristic is followed by the term $|\det D\beta_i|$, which measures the surface of a pixel from u projected to v_i. The larger the resolution of camera i, the bigger this surface, so that resolution sensitivity is properly handled. In addition, in Sec. 4.3.2.6 we have studied the effects of oversampling in the obtained weights. We have formalized the threshold proposed by Buehler *et al.* (2001), by stating that the proper handling of the oversampling allows not to penalize higher resolution cameras. In addition, when we take the sensor noise into account, we have shown that a higher resolution camera should be (marginally) preferred over an equal resolution image, because the supersampled sensor noise becomes smaller.

4.6.5 Equivalent Ray Consistency

"*Through any empty region of space, the ray along a given line-of-sight should be reconstructed consistently, regardless of the viewpoint position (unless dictated by other goals ...)*" Buehler *et al.* (2001). This is trivially satisfied by our framework, since the weights vary continuously when the novel view camera moves along an optical ray (through the continuous variation of the warp maps τ_i). The contribution of all views smoothly scales by taking into account the foreshortening effects.

4.6.6 Continuity

The *continuity* principle in IBR demands that the final rendered image varies continuously with the camera parameters of the original views. This implies that there are no seams at visibility boundaries between cameras, which may happen near the borders of the intersection of the field of view of each camera with the scene, or at depth discontinuities seen from each camera. The typical heuristic to enforce this form of continuity is to lower the contribution of a camera near a visibility boundary or the boundary of its field-of-view Raskar and Low (2002); Buehler *et al.* (2001). Our equations do not explicitly satisfy this property and the obtained weights do not fall to zero when approaching a visibility boundary. However, during the creation of our model, some observations pointed in that direction. Let us review them.

In Sec. 4.5.1.6 we saw that the *continuity* constraint near a visibility boundary could be enforced by the uncertainty values provided by the input. If the per pixel depth uncertainty σ_{z_i} is computed with the proposed geometric variations (Eq. 4.46), pixels near a visibility boundary have a higher geometric uncertainty, and their contribution to the reconstruction of the other rays is lowered. Only when reconstructing rays with a very small *angular deviation* its contribution may increase due to the fact that the high per pixel depth uncertainty σ_{z_i} is compensated with the low angular value (see Eq. 4.79).

While the per-pixel depth uncertainty can enforce the continuity along visibility boundaries of the scene, we found no evidence on why the contribution of a pixel at the boundary of the field-of-view should smoothly fall to zero. The *continuity* heuristic proposed by Buehler *et al.* (2001) reducing the contribution of an image along the visibility boundaries of the scene aims at reducing the seams in the transitions between the different images. A similar problem has been addressed in the literature of image stitching. For example, the method proposed by Levin *et al.* (2004) proposes to work in the gradient domain, in order to overcome the photometric inconsistencies between the images. Instead of focusing on how the contribution of an image could diminish along the visibility boundaries, future work could explore how the generative model could be extended to be applied to the gradient domain, and thus directly address the problem of the transition seams.

Although the per-pixel depth uncertainty fulfills the *continuity* desirable property near visibility boundaries, our equations have a fundamental problem that might violate it at any location. The term ∇u can strongly vary from one pixel to another, and thus the *intra-image smoothness* described in Raskar and Low (2002) is not fulfilled. As we saw in Fig. 4.18 artifacts are introduced. Is Sec. 4.5.7 we discussed possible leads to avoid the dependency on the latent image. However, since we claim to have a completely physics-based Bayesian formulation, any operation on the equations should be sustained by a physical explanation, which we are still missing, and is part of our future work.

Note that the prior term in the energy reduces the problems, most notably visual artifacts, which are due to not handling the continuity properly. However, a prior on the novel views cannot completely solve the continuity problem, which depends on the scene and camera geometry.

4.6.7 Real-Time

The final "*desirable property*" is for the method to be *real-time*. Our method is not yet real-time, mainly because of the computational complexity of the MAP estimate: 2 to 3 seconds are necessary to render a 768×768 image from 8 source images.

If super-resolution is not important, instead of solving the full MAP problem, it seems reasonable to use a direct method together with real-time regularization in the form of inpainting methods to obtain an acceptable result. As discussed in Sec. 4.5.7, rendering times could be reduced to nearly real-time. Moreover, as both the resolution algorithms and the hardware architectures are evolving quickly, much better performance can be expected in the next few years.

4.6.8 Balance Between Properties

One of the advantage of our method with respect to Buehler *et al.* (2001) is that the balance between the different properties is not handled by user-defined parameters, but implied from a formal deduction. Imagine a configuration with two cameras: one with low *minimal angular distance* but high *resolution sensitivity* change, and another with high *minimal angular distance* but low *resolution sensitivity* change. Which one should contribute more to the final image? In Buehler *et al.* (2001), the *angular distance* is preferred to the *resolution sensitivity* by a ratio of $1/0.05 = 20$ (Hallway dataset). In our equations, these variations are completely physics-based. An angular deviation of $\Delta\alpha$ between views is penalized proportionally to $\frac{1}{\sin^2 \Delta\alpha}$, due to the change in $\sigma_{g_i}^2$. A foreshortening effect or resolution difference causing an image scale factor s is penalized proportionally to $\frac{1}{s^2}$, due to the change in $|\det D\beta_i|$. The balance between these factors is properly handled by taking into account the sensor noise σ_s^2.

An exception is the weight λ, used in the prior term. Note that this is common in all work on image analysis based on Bayesian principles: since there is currently no meaningful way to obtain a prior distribution on the space of images, one needs to work with regularization by objective priors. Of course one could also use existing methods (Roth and Black, 2005) allowing to estimate this prior directly from the input images, thus obtaining a completely parameter-free model.

4.7 Summary of Contributions

The main contribution of this chapter is to establish the first formal link between the heuristics proposed in the recent decades for novel view synthesis, and the energy deducted by a physics-based generative model.

This model can be used to solve the generic problem which consists in generating a novel view from a heterogeneous set of input images, and a geometric description of the scene (called a geometric proxy), which can be either explicit (i.e. the estimated geometry of the 3D scene) or implicit (i.e. a set of correspondence maps between original views and the novel view).

Part of our contribution is the analysis of how the proposed model fulfills almost all the guidelines established by Buehler *et al.* (2001). The proposed generative model provides a formal description of the intuitive heuristics behind these guidelines. The key element to this unification is to take into account the error in the estimated geometric proxy when rendering a new image. We have extensively discussed how our physics-based model explains the reasons why some important heuristics were picked up in the first place. The theoretical benefits of the model outperform state of the art by overcoming its limitations. Moreover, the experiments conducted on synthetic and real images show that our method improves state of the art performance in terms of rendered image quality in the Lumigraph configuration and obtains state of the art performance in the Unstructured Lumigraph configuration.

We also discussed how future work can address the remaining issues of the proposed method. An important observation is that if the 3D reconstruction method or the 2D-2D image correspondence method provides not only depth estimates, but also the associated depth uncertainty, the image-based rendering method can benefit from this information to create better novel views. This should thus be a goal when developing new (implicit or explicit) reconstruction methods aimed at IBR.

5

The Stereoscopic Zoom

In this chapter, we explore two different possibilities to create stereoscopic shots with long focal lengths. Although the title of the chapter is "The Stereoscopic Zoom", technically, the word "zoom" only describes the possibility of a lens to change its focal length. Lenses are either labeled as "fixed focal length" or "zoom", independently of the magnitude of their focal length. However, in the cinema and television, most lenses equipped with a long focal length are zooms, because the cameraman needs to adjust the focal length to create the desired image frame. Thus, along the chapter we (ab)use the word "zoom" to refer to a long focal length.

To demonstrate the different possibilities to create a stereoscopic zoom, we focus on a simplified layout of the scene, which consists of a main subject of interest and a background farther away. It is a classic scenario where the zoom is used in 2D. An example shot arises in sports, where we want the closeup of the player when he concentrates just before the penalty kick, or when he just missed an easy goal with the hands on his head. Another example shot could arise in a live rock concert, where we would like to have a closeup shot of the singer at a sensitive moment, or a closeup of the guitarist when performing a solo. Because the physical cameras can not disturb the performance, their location is constrained. In our simplified configuration we assume that the cameras can not be "on the field", and must stay outside at a certain distance.

In Fig. 5.1 we illustrate a sketch of this layout representing a player on the field with the bleachers on the background. The distance between the cameras and the subject of interest and the background are z_s and z_b respectively. Our goal is to establish the position $\boldsymbol{c}_i$ of each actual camera together with its intrinsic parameters to render the stereoscopic images following the director's stereoscopic mise-en-scene. We constrain the points $\boldsymbol{c}_i$ on the $z = 0$ plane, and note $\boldsymbol{c}_i = (x_i, y_i, 0)$, where x_i is the horizontal displacement and y_i its height.

In order to generate stereoscopic images we need guidance on the nature of the shot we want to create. We focus on two different stereoscopic mise-en-scene. The first approach is guided by the intention to "get closer" to the scene, thus the

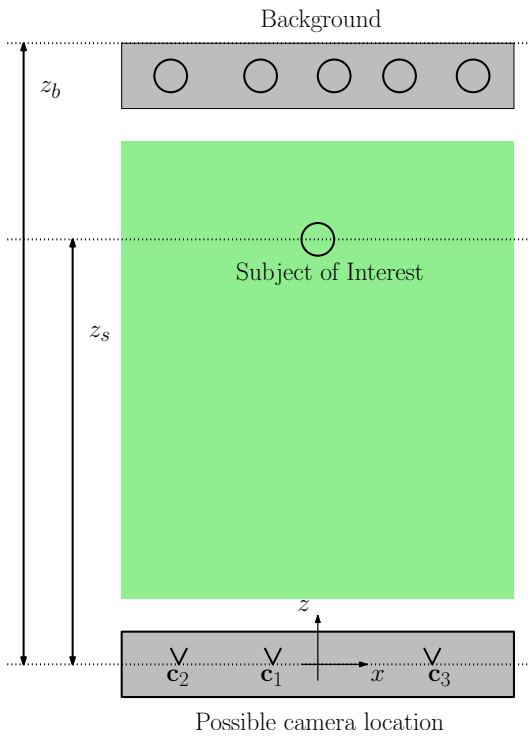


Fig. 5.1: *Simple layout of a scene: a main subject of interest, and a background. Actual cameras can only be placed outside the field. The distance between the actual camera location and the subject of interest is z_s. The distance between the actual camera location and the background is z_b.*

director places two virtual cameras on the field. Our goal is then to generate the images of the virtual views. The second approach is guided by the intention to add perspective deformations to the final stereoscopic images. The shot composition starts from a 2D frame and 3D annotations, which describe the depth of the scene elements and their roundness. In both approaches we use the IBR technique described in Sec. 4.3 to generate a pair of images fulfilling the desired properties.

5.1 Being On the Field!

5.1.1 The Mise-en-Scene

The first stereoscopic mise-en-scene would be the natural placement of the cameras if it were possible: the director would place them right on the field. The location is chosen together with the virtual filming configuration: the baseline b_v, the convergence distance H_v and the convergence window width W_v. As we saw in Chapter 3, the choice of this parameters is strongly dependent on the projection configuration (b', H', W').

To build the stereoscopic mise-en-scene the director has to decide on several artistic choices. We illustrate with an example how these choices are guided. The first choice is usually the depth of the subject in the cinema. It would be

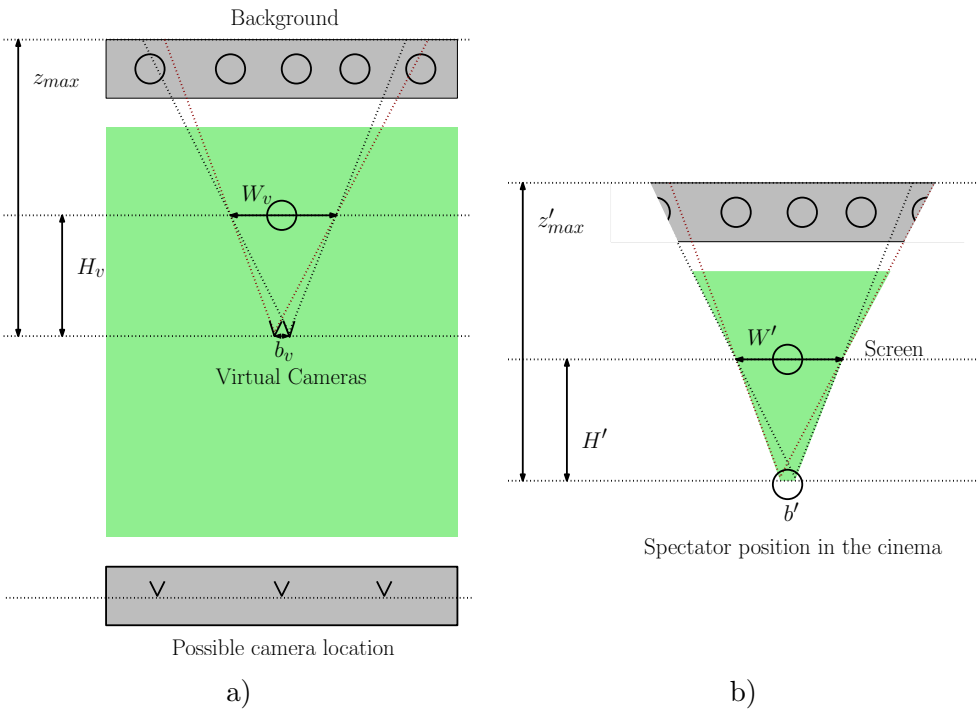


Fig. 5.2: *Top view of the acquisition and projection of the scene where the camera placement is not constrained. The director freely chooses to use the homothetic setup to keep a constant roundness factor. We illustrate the result of the stereoscopic mise-en-scene: a) scheme of the scene layout and the placement of the virtual cameras. b) scheme of the perceived depth from stereopsis by the audience in the projection room.*

recommended to choose its depth to be close to the screen, as it is the most comfortable stereoscopic viewing zone (see Fig. 2.4), but, of course, any variant is possible if the director desires a specific depth effect or depth transition. The size of the subject of interest in the image, i.e. the framing, establishes the width W_v. Once W_v is chosen, the director can for example decide to use the canonical setup to avoid ocular divergence (see Sec. 3.2.1.1). The baseline $b_v = b'\frac{W_v}{W'}$ is established. Let us recall that this configuration has the advantage to create a linear mapping of the depth between the cinema and the projection room, as well as a constant roundness $\rho(z) = \frac{W_v H'}{W' H_v}$ for all depths. The last choice is the convergence distance H_v, together with the camera placement. As we assumed that the convergence plane is set at the subject's depth, H_v is equal to the distance of the camera to the subject. With H', W' and W_v fixed, the choice of H_v defines the roundness of the shot $\rho(z) = \frac{W_v H'}{W' H_v}$. For example, to obtain a *natural* shoot with roundness $\rho = 1$ the director could decide to shoot with the homothetic setup (see Sec. 3.2.1.2):

$$\frac{b'}{b_v} = \frac{H'}{H_v} = \frac{W'}{W_v} \tag{5.1}$$

The roundness choice finishes the camera placement and the setting of all acquisition parameters. In Fig. 5.2 we illustrate the virtual camera placement as well as the

perceived depth from stereopsis. The focal length of the both cameras is $f_v = s\frac{H_v}{W_v}$, where s is the sensor width.

Let us remark, that the proposed virtual camera placement is only an example. For instance, we saw in Sec. 3.2.3 that if the target screen is not wide enough, the vergence-accommodation conflict may establish a stricter limit on the shooting baseline b_v than the *canonical setup*. We saw also that the director may choose to create an ocular divergence up to 0.5°. The stereoscopic mise-en-scene process is totally unconstrained. Our approach is generic with respect to the chosen parameters and applies to any virtual camera placement and stereoscopic configuration. Let us now study how the acquisition cameras should be placed in order to obtain the desired images.

5.1.2 The Quadri-Rig

To place the actual cameras, we use our proposed Bayesian IBR approach described in Sec. 4.3. The terms involved in the energy Eq. 4.26 guide our decisions to find the actual camera matrix $\boldsymbol{P}_i$. The image resolution is guided by $|\det D\beta_i|$ and $|\det D\beta_i|''$, and the optical ray angles are taken into account by the depth uncertainty term σ_{g_i} from Eq. 4.15. Once the camera positions and parameters are deduced, we analyze the obtained visibility m_i of the scene elements.

To establish the camera matrix of the actual cameras we want the contributing weight of the acquired image to be maximal when rendering the final image. For each individual camera, if we consider that it is the only one capturing the subject of interest, the magnitude

$$\frac{|\det D\beta_i|}{\sigma_s^2 + |\det D\beta_i|''(\sigma_{g_i}^2 \circ \beta_i)} \tag{5.2}$$

can be considered as a quality measure of how well the virtual image is rendered with the acquired image. An image with a higher weight would be preferred over another with a lower weight.

Note that another question would be how to place the camera in order to improve the virtual view if we already had a set of cameras. This is an interesting and difficult question, that we leave for future work. Moreover, to properly compute the camera matrix $\boldsymbol{P}_i$, one would need to have an estimate of the geometric shape of the subject of interest g_s and the background g_b. With those geometries, the camera position and focal length could be estimated as the values maximizing the resulting weight over all the surface elements $\boldsymbol{x}$ of the geometry, i.e. solving for

$$\arg\max_{\boldsymbol{P}_i} \int_g \frac{|\det D\beta(\boldsymbol{P}_i)|}{\sigma_s^2 + |\det D\beta(\boldsymbol{P}_i)|''(\sigma_{g_i}^2(\boldsymbol{P}_i) \circ \beta_i)} d\boldsymbol{x}. \tag{5.3}$$

We leave the investigation of how to optimize the camera position and parameters depending on the acquired geometry as future work.

In our work, we place the cameras one by one, and instead of jointly optimizing the position of the camera and their focal length with a given geometric proxy, we

proceed in two stages. First we consider both the subject and the background to be flat, and compute the focal length with respect to the virtual camera resolution f_v. We select the actual focal length so that $|\det D\beta_i| = 1$. Then we consider both the subject and the background to be punctual, and compute their position with respect to the value σ_{g_i}. We compute the camera position $\boldsymbol{c}$ minimizing σ_{g_i}, which, as we saw in Sec. 4.3.2.3, accounts for the minimal angular deviation. In all computations σ_s is considered constant and independent of the camera focal length and location.

All cameras are assumed to look in the z-direction, i.e. their rotation matrix $\boldsymbol{R}$ is the identity, and their principal point is adjusted depending on the camera location to obtain the final frame.

Because the scene can be roughly decomposed in two layers, we propose to use two actual cameras to generate each virtual view. To generate the left virtual view we propose to use a camera to acquire the subject of interest and another one to acquire the background. Symmetrically, to generate the right virtual view we propose to use a camera to acquire the subject of interest and another to acquire the background. We note the positions of the actual cameras $\boldsymbol{c}_j^k = (x_j^k, y_j^k, 0)$, and their focal lengths f_j^k, with $j \in \{l, r\}$ and $k \in \{s, b\}$. The subscripts l and r stand for *left* and *right* and the superscripts s and b stand for *subject* and *background*.

We name the resulting acquisition system the "Quadri-Rig".

5.1.2.1 Choosing the Focal Length

Our resolution goal can be written as $|\det D\beta_i| = 1$ or $|\det D\beta_i|'' = 1$. We could be tempted to acquire an image with a higher resolution in order to obtain a higher weight. For example, we could use cameras with a higher resolution, or, if the subject does not fill the whole width of the image, we could frame it closer, so that we capture it with a higher resolution.

Several considerations point against those decisions, both from a practical and theoretical point of view. The first one is that the resolution of the target image is usually the highest resolution we can actually capture. For example, if 4K (4096×2160) cameras are available for the acquisition, the target resolution will be most probably 4K. We do not want to assume that we can shoot with 4K cameras

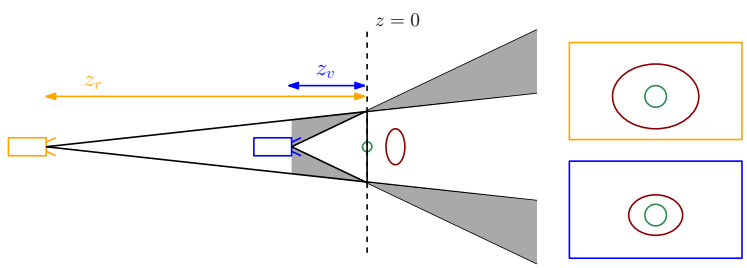


Fig. 5.3: *Two cameras with different focal lengths acquiring a green object with the same resolution. The object has the same size on both images. A red object farther away from the green object has a bigger image size in the camera with a longer focal length.*

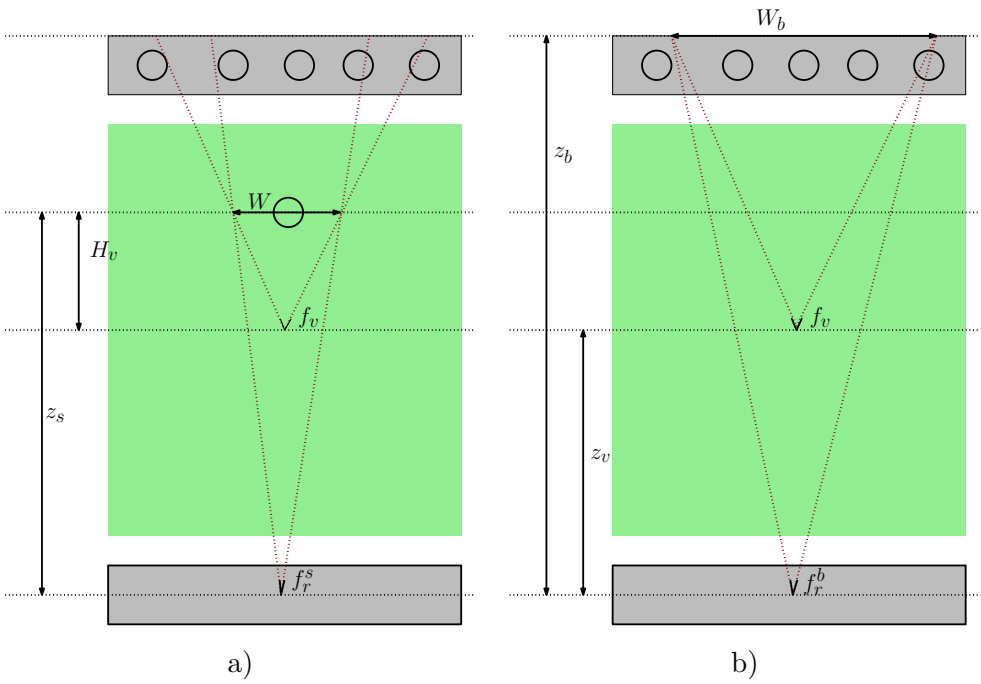


Fig. 5.4: *Acquiring the different parts of the scene with the same resolution as the virtual camera. f_v is the focal length of the virtual camera. a) The focal length f_r^s acquires the subject of interest at depth z_s with the same resolution as the virtual camera. b) The focal lenght f_r^b acquiring the background at depth z_b with the same resolution as the virtual camera.*

and render HD (1920×1080) images. The second consideration is that although it could be possible to frame the subject with a smaller size than W_v, the subject may almost fill the whole vertical extent of the image. Thus framing the subject closer could lead to the loss of some important part of the subject. The last consideration is that, as we saw in Sec. 4.3.2.6, the use of supersampling only marginally increases the contribution of the camera with a higher resolution by reducing its sensor noise. Thus, theoretically and practically, there is no need to capture the image at a higher resolution.

To obtain an image of a flat element with an equivalent resolution at a distance z_r with a focal length f_r and at a distance z_v with a focal length f_v, the relation between the focal lengths is

$$f_r = f_v \frac{z_r}{z_v}. \tag{5.4}$$

Note that this computation is only valid for a flat element at a punctual depth. Elements in front (or behind) this depth have an image size increase (or decrease) depending on the distances z_r and z_v as shown in Fig. 5.3. Because all cameras have the same sensor and the same resolution, the image size of an object can be directly translated into the acquired resolution of the object. The relation between the image size in pixels of an object at depth z in both images is given by the non linear function

$$w_r = w_v \frac{z_r}{z + z_r} \frac{z + z_v}{z_v}, \tag{5.5}$$

where w_r is the image size in pixels of the object in the actual camera and w_v is the image size in pixels of the object in the virtual camera.

Because of the symmetry in the configuration, both left and right cameras acquiring the subject of interest have the same focal length

$$f_s = f_v \frac{z_s}{H_v}, \tag{5.6}$$

and both left and right cameras acquiring the background also have the same focal length

$$f_b = f_v \frac{z_b}{z_b - z_v}. \tag{5.7}$$

We illustrate the subject and background focal lengths in Fig. 5.4. Note that if the acquired background is at infinity, then f_b is equal to f_v. If the subject of interest is at a different depth than the convergence distance of the virtual camera, the actual focal length of Eq. 5.6 should be of course adapted using this depth instead of H_v.

5.1.2.2 Choosing the Camera Positions

Our camera position goal can be written as the camera configuration minimizing the term σ_{g_i} from Eq. 4.15:

$$\underset{f,x,y}{\arg\min} \left(b * \left| (\nabla u \circ \tau_i) \cdot \hat{\sigma}_{z_i} \frac{\partial \tau_i}{\partial z_i} \right| \right). \tag{5.8}$$

The equation can be decomposed in three terms: $(\nabla u \circ \tau_i)$ accounting for the local variation in color of the target image, $\hat{\sigma}_{z_i}$ accounting for the geometric uncertainty and $\frac{\partial \tau_i}{\partial z_i}$ accounting for the angular deviation.

The angular distribution of the color gradient ∇u in an image can be measured, and does not have, in general, a uniform angular distribution (Torralba and Oliva, 2003). A dominant angle in the distribution, e.g. created by strip patterns or fences, would allow us to place the actual camera so that the epipolar lines with respect to the final view are orthogonal to the dominant angle. However, the angular distribution of the image may evolve over time, thus requiring the acquisition device not only to be aware of the image gradients, but also to adapt to them. Hence, we consider the camera placement to be independent of the first term $(\nabla u \circ \tau_i)$. The geometric uncertainty $\hat{\sigma}_{z_i}$ could be dependent or independent of the camera position and focal length, depending on the method to estimate the geometric uncertainty. For example, if the geometry is estimated with a depth range camera, the camera position and focal length do not affect $\hat{\sigma}_{z_i}$, whereas if the camera is used to compute the geometric uncertainty (see Sec. 4.5.1.3), then $\hat{\sigma}_{z_i}$ depends on the camera position and focal length. For the sake of simplicity, we assume the geometric uncertainty to be independent of the camera position and focal length and leave this line of joint estimation as future work. Hence, our goal to minimize σ_{g_i} is equivalent to minimize $\frac{\partial \tau_i}{\partial z_i}$. The generic warps τ_i from the actual camera to the virtual camera can be computed as described in Sec. 4.5.2.1. Then, we only need to minimize $\frac{\partial \tau_i}{\partial z_i}$

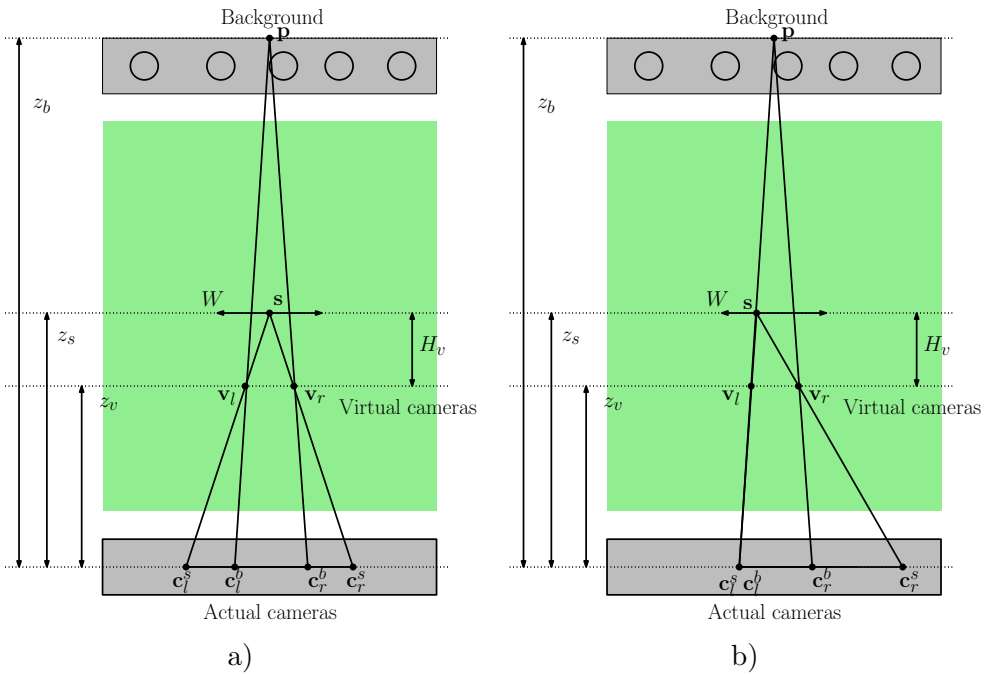


Fig. 5.5: *Diagram of the obtained baselines for the "Quadri-Rig". a) The actual cameras acquiring the subject are aligned with the position of the subject of interest* $\boldsymbol{s}$ *and the virtual cameras. The triangle defined by* $(\boldsymbol{s}, \boldsymbol{v}_l, \boldsymbol{v}_r)$ *and the triangle defined by* $(\boldsymbol{s}\ \boldsymbol{c}_l^s\ \boldsymbol{c}_r^s)$ *are homothetic. The cameras acquiring the background are aligned with the center of the background* $\boldsymbol{p}$ *and the virtual cameras. The triangle defined by* $(\boldsymbol{p}, \boldsymbol{v}_l, \boldsymbol{v}_r)$ *and the triangle defined by* $(\boldsymbol{p}\ \boldsymbol{c}_l^b\ \boldsymbol{c}_r^b)$ *are homothetic. b) When* $\boldsymbol{s}$, $\boldsymbol{p}$ *and* $\boldsymbol{v}_l$ *are aligned,* $\boldsymbol{c}_l^b$ *and* $\boldsymbol{c}_l^s$ *are equal.*

given by Eq. 4.57.

In order to avoid the need of a geometric estimate of the acquired scene element, we assume it to be punctual. It is then straightforward to see that the camera position minimizing $\frac{\partial \tau_i}{\partial z_i}$ is the one fulfilling the *epipolar consistency*: the camera position must be aligned with the optical center of the virtual camera and the position of the element to render. As explained in Sec. 4.3.2.3, as soon as the rays from both cameras are the same, the contribution of the input camera is maximal. Thus the position of the actual camera can be computed as the intersection of a 3D line and a plane. The line is defined by the optical center of the virtual camera and the 3D point representing the acquired scene element. The 3D plane is defined by the 0 depth plane, where the actual cameras can be placed.

The remaining question is how to choose the point representing the acquired scene element. A natural choice seems to select the center of the subject as its simplified 3D position. The center of the subject of interest can be easily approximated as the center of gravity of the 3D subject's points seen by the camera. Similarly, the center of the background can be chosen as the center of the background seen by both images.

Now that we have chosen the points to align the camera to, we compute the camera positions of the "Quadri-Rig". Let the left and right virtual camera positions

$\boldsymbol{v}_l$ and $\boldsymbol{v}_r$ be on the 0 height coordinate and at depth z_v:

$$\boldsymbol{v}_l = \begin{pmatrix} -\frac{b_v}{2} \\ 0 \\ z_v \end{pmatrix} \quad \text{and} \quad \boldsymbol{v}_r = \begin{pmatrix} +\frac{b_v}{2} \\ 0 \\ z_v \end{pmatrix}. \tag{5.9}$$

The subject of interest is at the location $\boldsymbol{s} = (x_s, y_s, z_s)$ and the background at a distance z_b from the cameras. The left camera position acquiring the subject of interest $\boldsymbol{c}_l^s$ is the intersection of the line defined by $\boldsymbol{s}$ and $\boldsymbol{v}_l$ and the 0 depth plane. Symmetrically, the right camera position $\boldsymbol{c}_r^s$ is given by the intersection of the line defined by $\boldsymbol{s}$ and $\boldsymbol{v}_r$ and the 0 depth plane. Their expressions are

$$\boldsymbol{c}_l^s = \begin{pmatrix} -\frac{b_v}{2}\frac{z_s}{H_v} - x_s\frac{z_v}{H_v} \\ -y_s\frac{z_v}{H_v} \\ 0 \end{pmatrix} \quad \text{and} \quad \boldsymbol{c}_r^s = \begin{pmatrix} +\frac{b_v}{2}\frac{z_s}{H_v} - x_s\frac{z_v}{H_v} \\ -y_s\frac{z_v}{H_v} \\ 0 \end{pmatrix}. \tag{5.10}$$

To compute the positions of the background actual cameras $\boldsymbol{c}_l^b$ and $\boldsymbol{c}_r^b$ we consider the point in the center of the background $\boldsymbol{p} = (0, 0, z_b)$. Thus we only have to set $x_s = 0$, $y_s = 0$, $z_s = z_b$ and $H_v = (z_b - z_v)$ in Eq. 5.10 and obtain

$$\boldsymbol{c}_l^b = \begin{pmatrix} -\frac{b_v}{2}\frac{z_b}{z_b - z_v} \\ 0 \\ 0 \end{pmatrix} \quad \text{and} \quad \boldsymbol{c}_r^b = \begin{pmatrix} +\frac{b_v}{2}\frac{z_b}{z_b - z_v} \\ 0 \\ 0 \end{pmatrix}. \tag{5.11}$$

In Fig. 5.5 we illustrate the camera placement of the four cameras creating the "Quadri-Rig".

Finally, the principal point of the actual cameras is chosen so that the camera frustums of the actual and virtual images intersect at the correspondent depth, i.e. background depth for the background cameras and subject depth for the subject cameras. Next we present the obtained camera matrices of the proposed "Quadri-Rig".

5.1.2.3 Camera Matrices

As the rotation matrix $\boldsymbol{R}$ is always the identity we do not detail it in every configuration. The intrinsic and extrinsic parameters of the *left virtual camera* are

$$\boldsymbol{K}_l^v = \begin{pmatrix} f_v & 0 & -\frac{b_v}{2} \\ 0 & f_v & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{t}_l^v = \begin{pmatrix} +\frac{b_v}{2} \\ 0 \\ -z_v \end{pmatrix}. \tag{5.12}$$

The intrinsic and extrinsic parameters of the *right virtual camera* are

$$\boldsymbol{K}_r^v = \begin{pmatrix} f_v & 0 & +\frac{b_v}{2} \\ 0 & f_v & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{t}_r^v = \begin{pmatrix} -\frac{b_v}{2} \\ 0 \\ -z_v \end{pmatrix}. \tag{5.13}$$

The intrinsic and extrinsic parameters of the *left actual camera acquiring the subject* are

$$\boldsymbol{K}_l^s = \begin{pmatrix} f_s & 0 & -\frac{b_v}{2}\frac{z_s}{H_v} - x_s\frac{z_v}{H_v} \\ 0 & f_s & -y_s\frac{z_v}{H_v} \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{t}_l^s = \begin{pmatrix} +\frac{b_v}{2}\frac{z_s}{H_v} + x_s\frac{z_v}{H_v} \\ +y_s\frac{z_v}{H_v} \\ 0 \end{pmatrix}. \tag{5.14}$$

The intrinsic and extrinsic parameters of the *right actual camera acquiring the subject* are

$$\boldsymbol{K}_r^s = \begin{pmatrix} f_s & 0 & +\frac{b_v}{2}\frac{z_s}{H_v} - x_s\frac{z_v}{H_v} \\ 0 & f_s & -y_s\frac{z_v}{H_v} \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{t}_r^s = \begin{pmatrix} -\frac{b_v}{2}\frac{z_s}{H_v} + x_s\frac{z_v}{H_v} \\ -y_s\frac{z_v}{H_v} \\ 0 \end{pmatrix}. \tag{5.15}$$

The intrinsic and extrinsic parameters of the *left actual camera acquiring the background* are

$$\boldsymbol{K}_l^b = \begin{pmatrix} f_b & 0 & -\frac{b_v}{2}\frac{z_b}{H_v} \\ 0 & f_b & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{t}_l^b = \begin{pmatrix} +\frac{b_v}{2}\frac{z_b}{z_b - z_v} \\ 0 \\ 0 \end{pmatrix}. \tag{5.16}$$

The intrinsic and extrinsic parameters of the *right actual camera acquiring the background* are

$$\boldsymbol{K}_r^b = \begin{pmatrix} f_b & 0 & +\frac{b_v}{2}\frac{z_b}{H_v} \\ 0 & f_b & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{t}_r^b = \begin{pmatrix} -\frac{b_v}{2}\frac{z_b}{z_b - z_v} \\ 0 \\ 0 \end{pmatrix}. \tag{5.17}$$

5.1.2.4 Optimal Baselines

We deduced each camera location and focal length independently of the other cameras using our findings from Sec. 4.3. However, the left and right cameras acquiring the background, as well as the left and right camera acquiring the subject, can be analyzed as two stereoscopic pairs. The deduced baseline to capture the subject of interest is

$$b_s = b_v \frac{z_s}{H_v}. \tag{5.18}$$

Although the position of the cameras depends on the subject position x_s and y_s, the obtained baseline does not. The terms x_s and y_s cancel out when we compute the differences between $\boldsymbol{c}_l^s$ and $\boldsymbol{c}_r^s$. Most interestingly, the same baseline was obtained in Eq. 3.98 as we studied which baseline allowed to obtain a roundness factor ρ in the projection room. If we only project the part of the images with the subject of interest in the projection room, the audience will perceive the subject at the depth of the screen with the desired roundness factor ρ.

The baseline obtained for the background cameras is

$$b_b = b_v \frac{z_b}{z_b - z_v}. \tag{5.19}$$

As the focal lengths are different from the ones of the virtual cameras, it is not straightforward to see if ocular divergence happens. If the background of the scene is exactly a plane at depth z_b, the acquired images by the background cameras are, by construction, exactly equal to the images acquired by the virtual cameras (up to non-lambertian deviations). Thus the perceived depth in the projection room is the same whether we use one pair of images or the other, and ocular divergence only arises if the director decided to. Moreover, the roundness factor of the acquired background is the same as the roundness obtained with the virtual images. Although, as the acuity of the depth perception decreases with the distance of the scene elements (Howard and Rogers, 2008), it is questionable if the preservation of the roundness in the background should be a goal in itself.

Let us note that a "roundness only" reasoning does not establish an actual constraint on the position of the cameras, it only constrains their relative position, the baseline. As we saw, the camera position acquiring the subject of interest depends on the subject location $\boldsymbol{s}$. It may happen that $\boldsymbol{s}$ and $\boldsymbol{p}$ are aligned with a virtual camera, as we illustrate in Fig. 5.5b. Then, both left, or both right cameras have the same location. Physically, this is not a problem, as the use of a mirror rig allows to place two cameras with the same optical center. From a practical point of view, to change the camera location depending on the position of the subject in the frame may result in a too complex mechanical system. The subject should be detected and the camera position aligned consequently. A reasonable simplification is to fix the position of the point $\boldsymbol{s}$ at the center of the image, so that the long focal length cameras do not have to move as the subject moves in the image.

5.1.2.5 Visibility

Even in our simplified layout of the scene, four different kinds of occlusions may arise. The subject of interest naturally occludes some regions of the background. We name this kind of occlusion *subject-background*. The subject of interest may also occlude some parts of itself, e.g. an arm near the face can partially occlude the face. We name this kind of occlusion *subject-subject*. Similarly, the background may also occlude some parts of itself, e.g. a member of the crow in front of another. We name this kind of occlusion *background-background*. The last visibility issue may be created if an intruder enters the field of view of the actual cameras and we name

this occlusion type *intruder-scene*.

Subject-Background Occlusion A background camera needs to acquire the region of the background visible by the virtual camera. However, as the actual camera position is farther away, the subject of interest can occlude some regions of the background.

Let us first assume that the acquired subject of interest is convex. In Fig. 5.6a, we mark with a green rectangle the region of the background visible in the virtual camera and with a blue rectangle the region of the background visible in the actual camera. If the background camera is aligned with the subject and the virtual cameras, then the actual cameras always capture all of the needed background. However, as we illustrate in Fig. 5.6b, if the background camera is not aligned with the subject and the virtual camera, some part of the needed background may not be present in the source images.

To ensure the visibility of the background, the actual cameras are restrained to a visibility region. The area of the subject, together with the virtual camera position define a volume. The intersection of this volume with the actual camera plane defines the visibility region, as we illustrate in Fig. 5.6c with a pink rectangle. Any camera on this region captures all of the needed background, and its size depends on the size of the subject, the virtual convergence distance H_v and the distance z_v between the virtual cameras and the actual ones.

Note that although we illustrate these regions only for the x-coordinate, the same scheme applies for the y-coordinate. In most cases, the vertical visibility is not an issue, as the subject of interest covers the vertical center of the image. However, there are scenarios where this constraint should not be neglected, like for example, when we acquire a flying bat. If the animal is not framed on the vertical center of the image, some parts of the needed background are not acquired by the actual background cameras. To avoid background visibility issues, the operator should always keep the animal vertically centered on the image.

If the subject is not convex, the entire needed region of the background may be impossible to capture with a single camera. In Fig. 5.6d we illustrate the problem. The non-convex subject is represented by two elements. The space between both elements and the virtual camera focal length defines a background region which needs to be acquired. The size of this region is w_v and depends on the size of the gap between the scene elements Δ, the depth z_s and the background depth $z_b - z_v$:

$$w_v = \Delta \frac{z_b - z_v}{z_s - z_v}. \tag{5.20}$$

The size of the background region that can be acquired with the actual camera through both elements is

$$w_r = \Delta \frac{z_b}{z_s}. \tag{5.21}$$

A single camera at a farther distance and with a longer focal length can not acquire

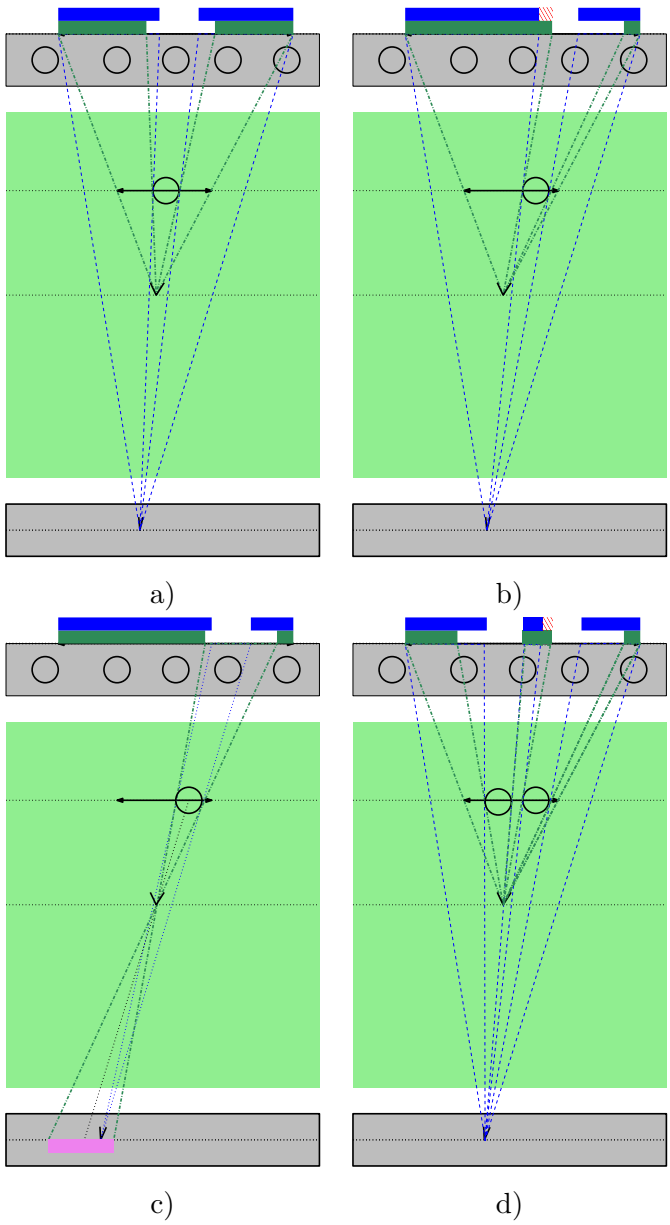


Fig. 5.6: *Subject-background occlusion. a) The virtual camera, the subject and the actual camera are aligned. The needed background areas for the virtual camera (green regions) are correctly captured by the actual camera (blue regions). Blue rectangles overlap the green ones. b) The subject moves and the virtual camera, the subject and the actual camera are not aligned anymore. The needed background areas for the virtual camera (green regions) are not completely captured by the actual camera (blue regions). A red zebra rectangle shows the missing area. c) The alignment constraint can be relaxed depending on the size of the subject. The borders of the subject and the virtual camera define two rays. Any camera inside those rays (purple regions) completely acquires the needed background areas. As soon as the actual camera is not on the regions, some background elements are occluded by the subject. d) If the acquired subject is not convex, the central part of the background needed by the virtual camera (green region) can not be acquired by a single camera. The center blue region in the background is always smaller than the center green region.*

the needed region trough the space between both scene elements, because

$$(1 - \frac{z_v}{z_s})\frac{z_b}{z_b - z_v} < 1 \; \forall z_s \in (z_v, z_b). \tag{5.22}$$

Subject-Subject and Background-Background Occlusions Similarly to the *subject-background* occlusions, if the subject or the background are not convex, self occlusions may arise. For example, an arm of the subject of interest might occlude its face. In this case, the problem can again be illustrated with Fig. 5.6d. We just need to consider the arm as the subject of interest and the face as the background. If the arm is not aligned with the virtual camera and the center of the face, some region of the face needed in the virtual image can not be acquired by the actual camera. The same reasoning applies to *background-background* occlusions.

Intruder-Scene Occlusion The last occlusion scenario we need to consider is the *intruder-scene* occlusion. In our approach we propose to shoot elements of the scene with cameras that are far away from the subject and the background. We implicitly assume that the space in-between the actual cameras and the virtual cameras is empty. While this assumption is not very restrictive for the actual cameras acquiring the subject of interest, it may be restrictive for the cameras acquiring the background. The shorter focal lengths of the background cameras may require an important free volume around the subject of interest. In Fig. 5.7 we illustrate the regions of the scene which should ideally be empty. If an obstacle enters these frustums, the actual cameras can not capture the desired part of the scene to be rendered.

To quantify the required free space, let us consider W_s, the width of the union of both camera frustums acquiring the background at the depth of the subject of interest. We illustrate W_s in Fig. 5.7b, which is given by the expression

$$W_s = W\frac{(z_b - z_v)}{H_v}\frac{z_s}{z_b} + b_v\frac{z_b - z_v}{H_v}. \tag{5.23}$$

To get an idea of the magnitude, let us evaluate the function with a numerical example. Let us assume $z_b = 50$m, $z_v = 25$m, the convergence width $W = 2$m and the virtual convergence distance $H_v = 3$m. We consider b_v to be small (in the order of cm) and thus ignore the additive term. Then, the width of the frustum is 9.33m. If we are acquiring a football player, the nearest player to it should be at least 4.5m away. While this constraint may be fulfilled in a sports settings where most of the space in-between the subject and the cameras is free, it may not be fulfilled in a shot in a forest in the wild.

Visibility Summary If the subject of interest is convex and at the center of the image, and the background is also convex, the proposed "Quadri-Rig" does not suffer from any occlusion other than *intruder-scene*. However, if the background is not convex, then *background-background* occlusions may arise. Moreover, if the

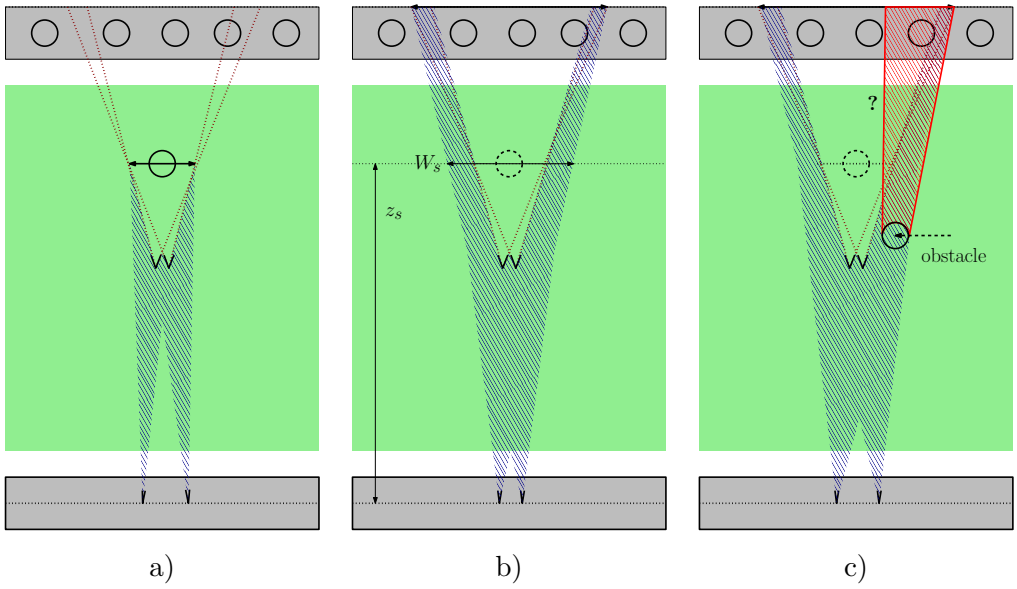


Fig. 5.7: *a) The actual cameras capture the subject with the same resolution as the virtual cameras. The zebra area must be empty to ensure the visibility of the subject. b) The actual cameras capture the background with the same resolution as the virtual cameras. The zebra area must be empty to ensure the visibility of the background. The width of the union of both camera frustums at depth z_s is W_s. c) An obstacle enters the field of view of the actual cameras. A large region of the background is not visible anymore.*

subject is convex but not on the center of the image, *subject-background* occlusion regions may also arise. Finally, if the subject is not convex, then *subject-background* or *subject-subject* occlusions may arise. In Sec. 5.1.4 we discuss how the visibility problem can be addressed.

5.1.3 Proof of Concept

In this section we present a proof of concept of the proposed approach. As we saw in Sec. 4.4.2 the final quality of the rendered images depends on the precision of the geometric proxy. Thus we created two datasets, a synthetic one where the ground truth geometry is available, and a second dataset with images from a real-world scene.

We name the synthetic dataset the *blender lego* dataset, which consists of 6 images, the 3D model used to render the images and the camera positions and parameters. The left and right virtual images were rendered at the desired virtual positions $\boldsymbol{v}_l$ and $\boldsymbol{v}_r$. These images were used as ground truth for comparison with the rendered images. The other four images of the dataset correspond to the "Quadri-Rig" configuration: two images of the cameras acquiring the subject and two images of the cameras acquiring the background. In our scene the subject of interest was placed at the center of the images. All images were rendered at HD resolution: 1920×1080. The real-world scene dataset is called *real lego* and contains the same 6 images: the two virtual views and the four "Quadri-Rig"

images. In addition, 6 other images were used to calibrate the camera positions and to reconstruct a geometric proxy using the pipeline described in Sec. 4.5.1. The resolution of the real-world images is 2376×1548. In Fig. 5.8 we illustrate the *blender lego* dataset and in Fig. 5.9 we illustrate the *real lego* dataset.

5.1.3.1 Results

In Table 5.1 we present the PSNR and DSSIM computed values. Because of the low number of images, all methods yield very similar results. This result is coherent with the results obtained in Pujades and Devernay (2014). The difference in the blending weights has no significant impact on the rendered images when few images are used in the blending. The high PSNR and DDSIM values obtained with the synthetic dataset show that the rendered images at visible locations are accurate. In Fig. 5.10 we reproduce the full resolution images rendered by the different methods for the synthetic dataset. However, because the background of the scene is highly non-convex, large regions visible in the virtual views are not acquired by any of the four actual cameras of the "Quadri-Rig". In Fig. 5.11 we show closeups of these regions.

For the real-world image dataset, the computed PSNR and DSSIM values shown in Table 5.1 are very low. In Fig. 5.12 we reproduce the full resolution images rendered by the different methods. The poor 3D reconstruction obtained from the input images creates important artifacts in the rendered images that we illustrate in Fig. 5.13.

5.1.4 Quadri-Rig Discussion

We now discuss the obtained camera configuration, with its advantages and limitations.

5.1.4.1 Camera Position Dependency on the Rendering Method

The "Quadri-Rig" camera configuration was deduced in order to maximize the quality of the rendered virtual images following the equations obtained in our IBR

	Blender Lego Dataset				Real Lego Dataset			
	left image		*right image*		*left image*		*right image*	
ULR	33.93	288	33.94	290	**17.12**	**882**	**16.94**	**889**
SAVS	**34.03**	**286**	**34.04**	**288**	**17.12**	**882**	16.91	**889**
Proposed	34.00	287	34.02	**288**	17.09	884	16.90	891

Table 5.1: *Numerical results for synthetic and real-world datasets. We compare our method to Wanner and Goldluecke (2012) (SAVS) and Buehler* et al. *(2001) (ULR). For each rendered image, the first value is the PSNR (bigger is better), the second value is DSSIM in units of* 10^{-4} *(smaller is better). The best value is highlighted in bold.*

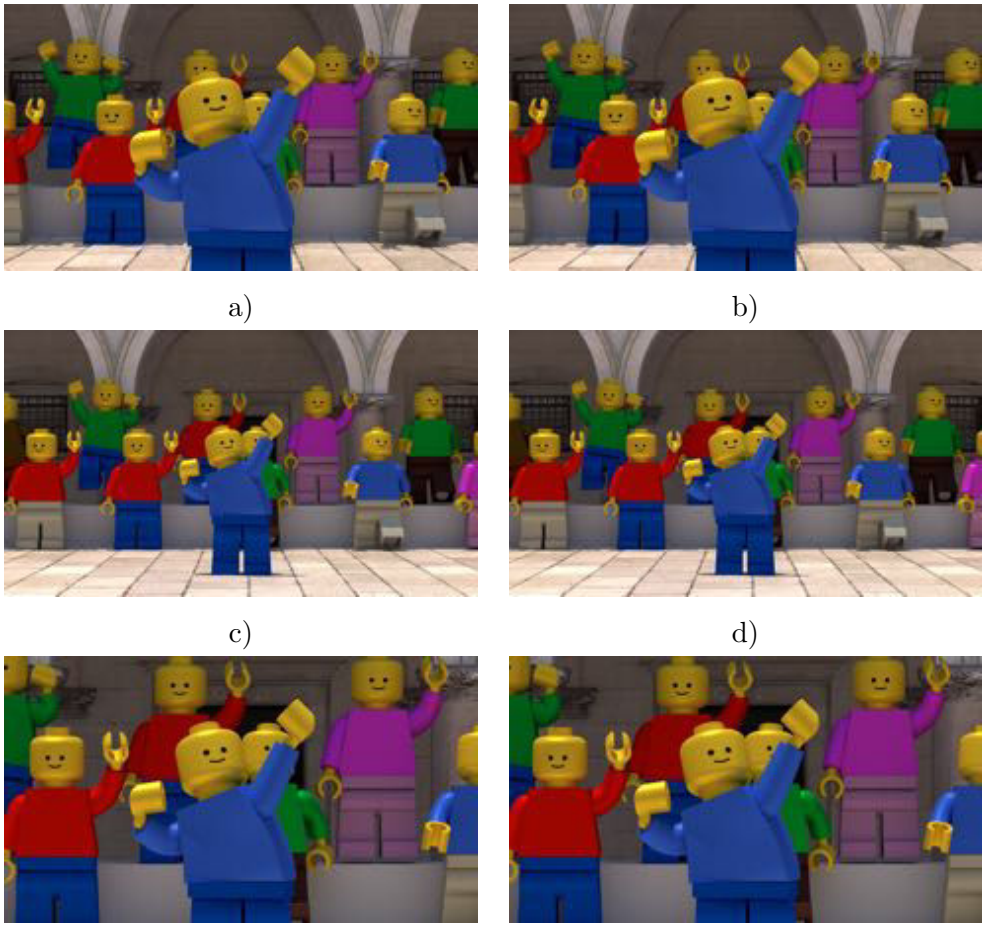


Fig. 5.8: *The "blender lego" dataset: a) and b) are the left and right images of the virtual stereoscopic pair. c) and d) are the left and right rendered images with the background cameras of the "Quadri-Rig". e) and f) are the left and right rendered images with the subject of interest cameras of the "Quadri-Rig".*

approach. In our study, as the computation relies on a punctual measure at an (arbitrary) point of the image, the camera position problem is equivalent to the proper handling of the *epipole consistency* desirable property from Buehler *et al.* (2001). The obtained camera configuration maximizes the quality of the rendered images not only with our IBR method, but with any other method fulfilling this property, like for example the methods proposed by Buehler *et al.* (2001), Levoy and Hanrahan (1996) or Gortler *et al.* (1996). Future work should study how the camera position varies for the different rendering methods if a geometric proxy is available for the subject of interest and the background.

Let us recall that the weight equations obtained in Wanner and Goldluecke (2012) only take into account the resolution sensitivity. In addition, as we saw in Sec. 4.3.2.6, we have no evidence pointing out that the resolution weight should be thresholded: the higher the resolution of an acquisition camera, the higher its

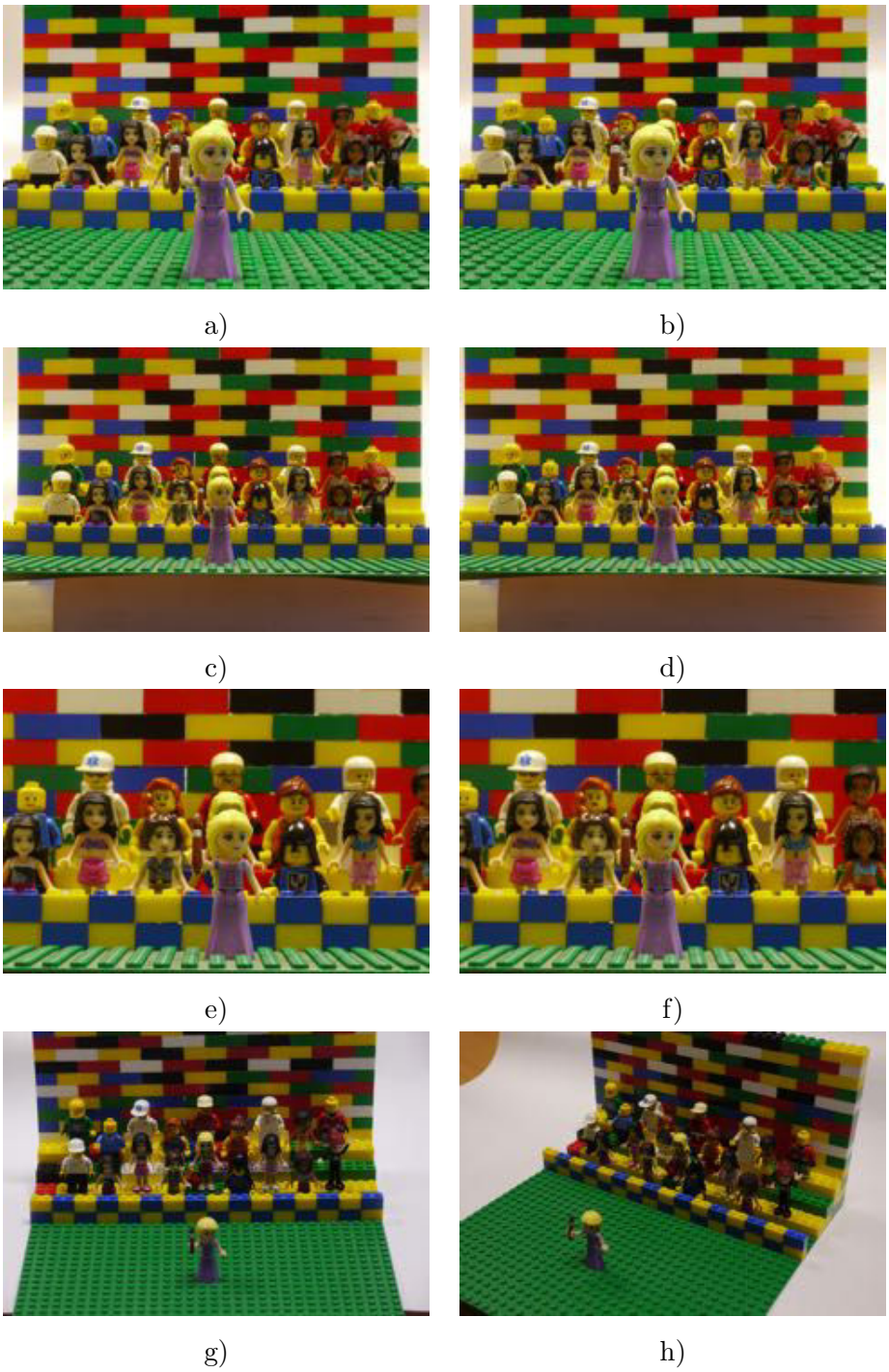


Fig. 5.9: *The "real lego" dataset: a) and b) are the left and right images of the virtual stereoscopic pair. c) and d) are the left and right acquired images with the background cameras of the "Quadri-Rig". e) and f) are the left and right acquired images with the subject of interest cameras of the "Quadri-Rig". g) an h) are two of the six auxiliary images used to calibrate the dataset and create the 3D reconstruction.*

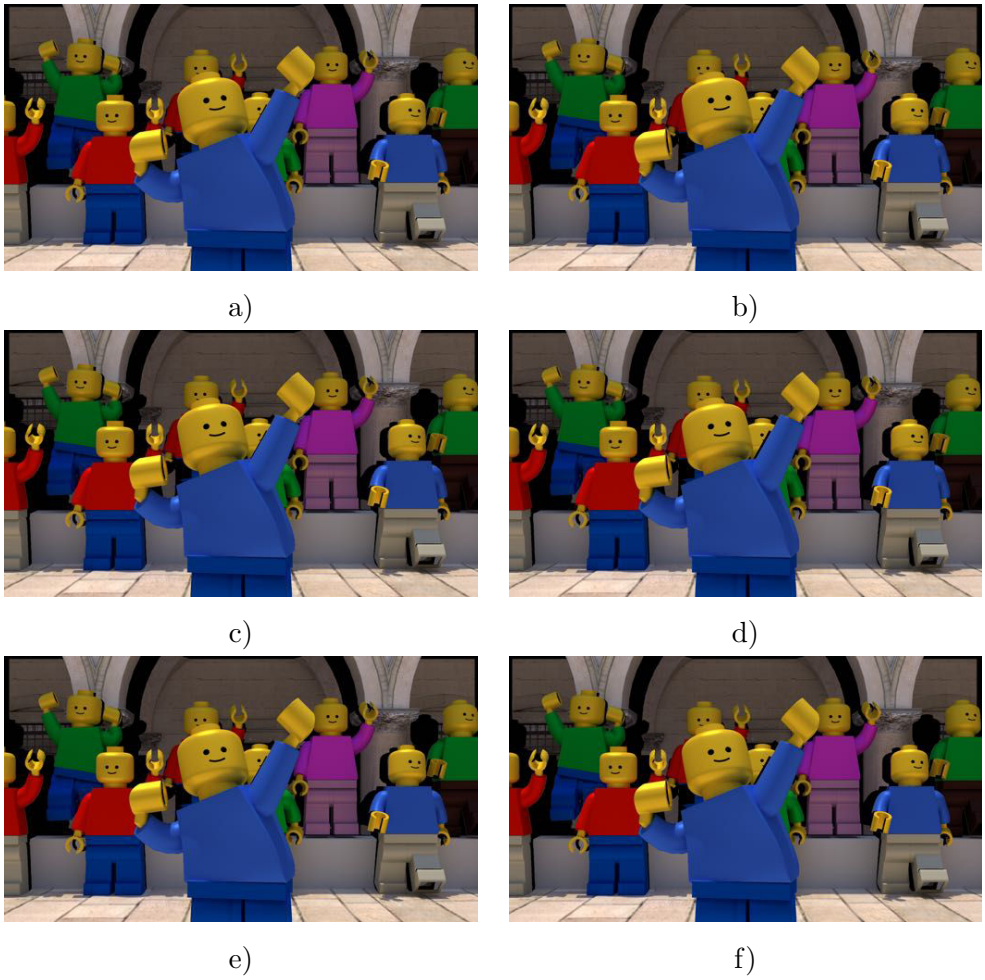
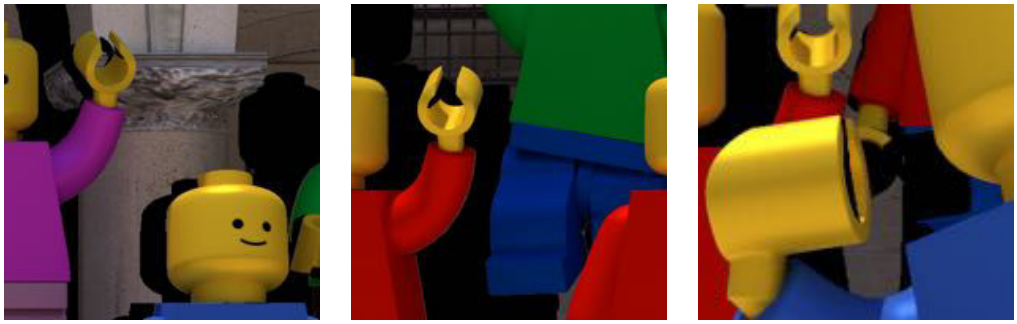


Fig. 5.10: *Rendered images obtained with the different methods with the "blender lego" dataset. a) and b) left and right images rendered with Buehler* et al. *(2001). c) and d) left and right images rendered with Wanner and Goldluecke (2012). e) and f) left and right images rendered with our method. Because few images (4) were used in the rendering, the image quality among the different methods is very similar.*

weight. This framework's equations do not provide a practical solution to the problem of how to chose the actual camera placement and their focal length. One should use the highest resolution camera available, together with the longest focal length possible. Moreover, as the resolution sensitivity desirable property does not constraint the camera location, any camera position could be used. Thus we would have no way to deduce the actual camera positions.

5.1.4.2 Optimal Camera Properties

In Sec 5.1.2.4 we observed an interesting result. The deduced camera positions with our Bayesian IBR approach establish the same baseline as the one obtained with a roundness factor reasoning presented in Sec. 3.5. In addition, the resolution



a) b) c)

Fig. 5.11: *Closeups of the rendered views. As the background is highly non convex, large areas needed for the virtual cameras are not acquired by the actual cameras. a) and b) The black pixels behind the crowd are not acquired by any actual camera. c) The black pixels in the inner side of the hand of the subject of interest are not acquired by any actual camera.*

sensitivity in our approach, establishes the acquisition focal length. If the element of the scene is exactly a plane, the acquired images by the actual cameras are, by construction, exactly equal to the images acquired by the virtual cameras (up to non-lambertian deviations). Then the perceived depth of the element is exactly the same for both pairs of images. Most interestingly, the resolution sensitivity property, creates an equivalent perceived depth.

5.1.4.3 Visibility Issues

The main limitation of the proposed approach is the visibility issue. As we described in Sec. 5.1.2.5, multiple kinds of occlusion may arise. The subject and the background may be self-occluded, the subject may further occlude the background, and an obstacle may even occlude the subject and the background. As we illustrated in Fig. 5.11, both final rendered images contain important regions where no information is available. Moreover, to avoid further occlusions, the cameraman should frame the subject of interest at the center of the image, which results in an important limitation of the approach. The frame is one of the most important choices of a director and should not be constrained. Thus potentially, even more occluded regions may appear in the final images.

A possible solution to render the occluded regions is to use other cameras to acquire the missing texture. For example, in the setting proposed by Hilton *et al.* (2011), we could expect that one of the multiple cameras could provide texture information of the occluded region. Of course this solution is not straightforward, as registering wide baseline cameras is not easy, and a quite accurate geometric reconstruction is needed to warp the acquired texture. While this approach may be feasible to render occluded regions of the background, it might be of little help to acquire the parts of the subject suffering from subject-subject occlusions.

In regions where no additional texture information is available, then an inpainting algorithm should be used to fill the empty regions of the rendered image.

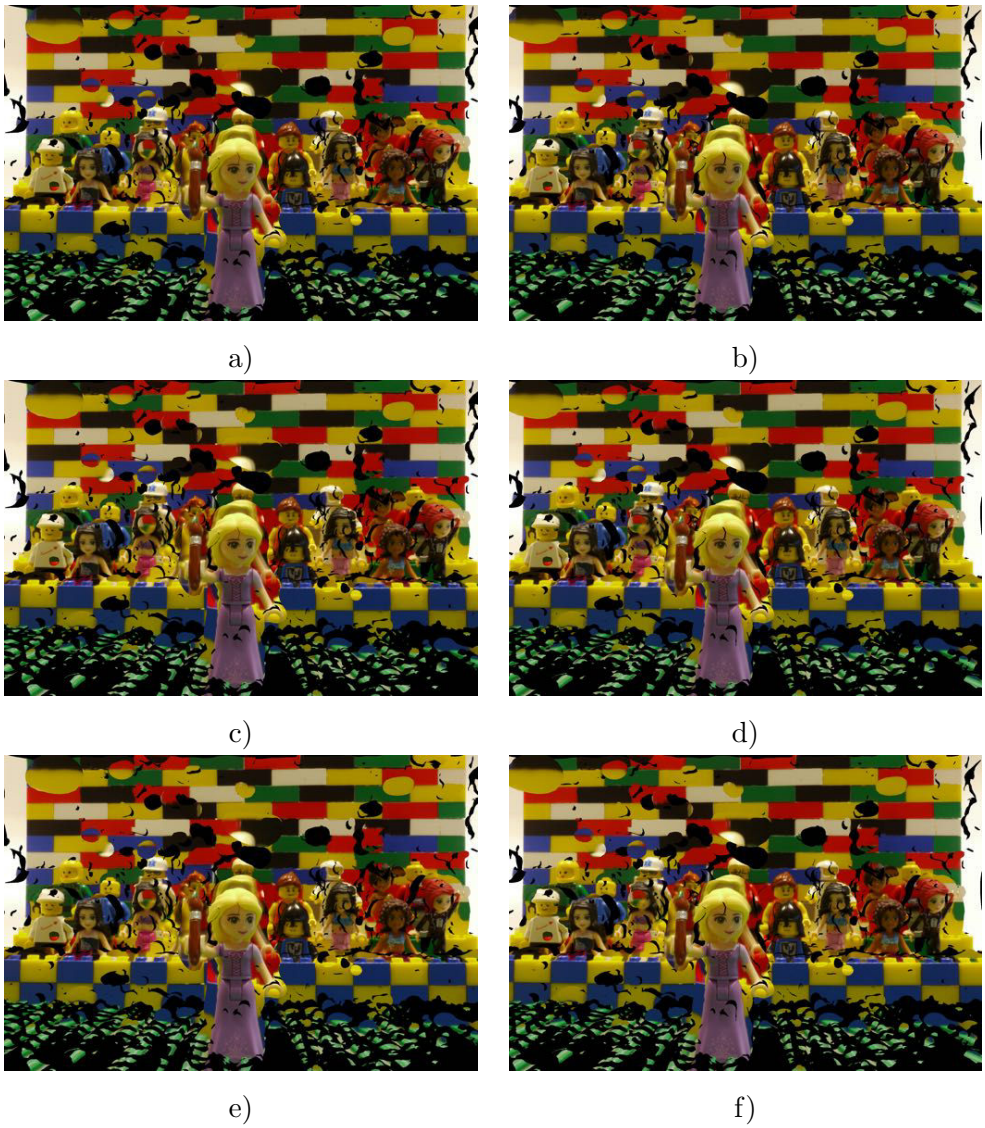


Fig. 5.12: *Rendered images obtained with the different methods with the "real lego" dataset. a) and b) left and right images rendered with Buehler* et al. *(2001). c) and d) left and right images rendered with Wanner and Goldluecke (2012). e) and f) left and right images rendered with our method. Because of the poor geometric proxy important artifacts are visible in the rendered images.*

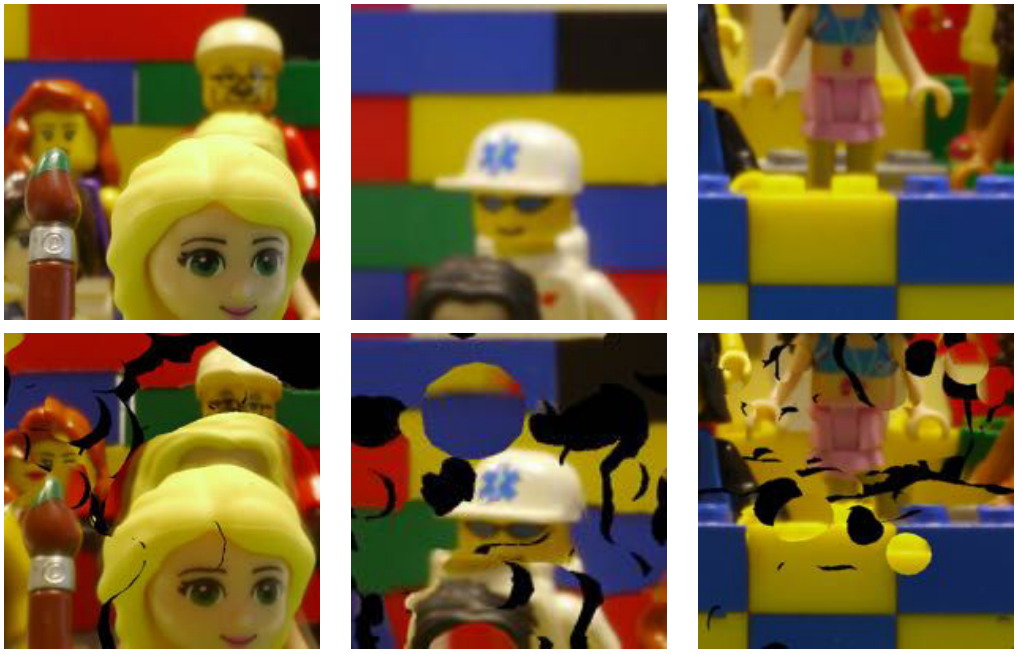


Fig. 5.13: *Closeups of the rendered views of the "real lego" dataset. In the top row we show closeups of the ground truth acquired virtual image. In the bottom row we show closeups of the rendered virtual image. As the 3D reconstruction is very poor, important visual artifacts arise.*

5.2 Distort the World!

In this section we present a second possibility to create stereoscopic shots with long focal lengths. Our approach is guided by a different stereoscopic mise-en-scene, which is inspired by the 2D to 3D conversion methods. The director establishes depth and roundness constraints on the scene elements. Then we study how these constraints translate into potentially multiple acquisition settings. Once the images are acquired, they have to be combined to obtain a pair of images creating the desired stereoscopic effect. Finally we present a proof of concept and discuss the obtained configuration and results.

5.2.1 The Mise-en-Scene

The first thing the director needs to chose is the 2D frame of the image. To do so, the director places a camera on the possible locations and adjusts the focal length to frame the subject of interest. Then, for each relevant element of the scene, the director specifies the expected perceived depth in the projection room. This depth description is usually done in disparity values, either in *pixel* units or in *percentage* of the image width. The director is free to choose any units, as *pixels* can be easily be converted into *percentage* and vice-versa if the resolution of the image is available. In addition to the depth, the director's description may also specify the expected *roundness factor* of each element.

This stereoscopic mise-en-scene approach is already used in actual 2D to 3D

conversion of feature films (Neumann, 2011). In Fig. 5.14 we illustrate an example of an initial 2D frame and its depth annotated version. These images provide guidance to the artists to create the full depth map.

At the time of the stereoscopic mise-en-scene the director is most probably unaware of the actual depth of the scene. However, the provided depth description establishes constraints relating the actual depth z_e of a scene element and its expected disparity d'_e (or perceived depth z'_e) at the projection room. These constraints can be written as $z'(z_e) = z'_e$. Similarly, a roundness description of the element establishes a constraint $\rho(z_e) = \rho_e$. Thus the depth annotated 2D frame can be translated into a set of constraints on the perceived depth function $z'(z)$ (Eq. 3.33) and the roundness factor function $\rho(z)$ (Eq. 3.62). As we still are in the acquisition stage, a natural question arises: which cameras allow to acquire the scene elements with the desired properties?

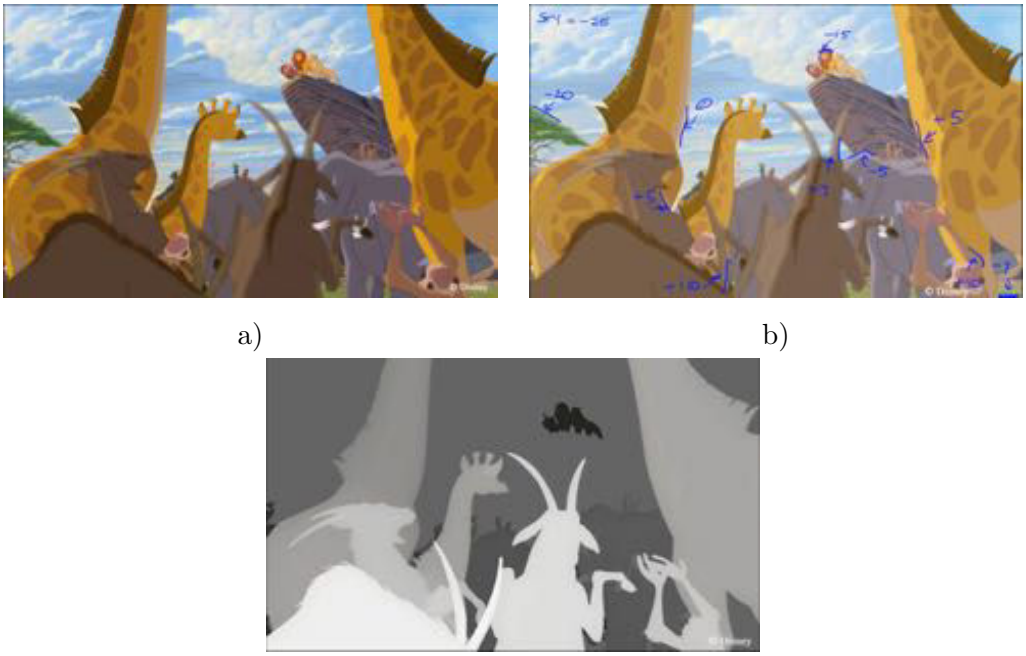
5.2.2 The Multi-Rig

Our goal now is to translate the depth and roundness factor constraints into a (potentially) multi-view acquisition device. As we saw in Sec. 3.2, when we acquire a scene with a pair of rectified cameras for a fixed target projection configuration (b', H', W'), the depth perception function $z'(z)$ from Eq. 3.33 has three degrees of freedom: the convergence window width W, its distance H, and the acquisition baseline b. If the director specifies more than three constraints, then we have an overdetermined system.

In our case, the focal length of the first camera has been chosen by the director to create the image frame. As we saw in Sec. 5.1.2.1, to acquire an element of the scene with the same resolution with two different cameras, their focal lengths have to follow the relationship of Eq. 5.4. In our case, as all cameras are at the same distance of the element, all focal lengths have to be equal. It follows that the ratio $\frac{H}{W}$ is the same for all acquisition setups, and a first constraint of our system is established. Only two degrees of freedom are left, and each couple of constraints establishes an optimal camera setup to acquire the element.

One depth constraint $z'(z_e) = z'_e$ together with a roundness constraint $\rho(z_e) = \rho_e$ fully constrain the acquisition setup. Similarly, if the convergence distance is kept constant for all constraints ($z'(H) = H'$) then only a depth or roundness constraint fixes the acquisition baseline. Hence, we may potentially need as many acquisition cameras as constraints. Although theoretically one could add a constraint on the roundness of an element but not on its depth, we can not imagine a case where a director would do that, as the whole process of the stereoscopic mise-en-scene is based on how the elements are placed in depth.

Let us recall, that in Sec. 5.1.2.4 we observed that the obtained optimal baseline to acquire the roundness of an element was the same baseline obtained according the our IBR approach. Hence, both our IBR approach and a roundness factor reasoning lead to the same baselines.



a) b)

c)

Fig. 5.14: *Images reproduced from Neumann (2011). The original image a) is annotated by the director with pixel values describing the expected disparity in the final shot b). Artists model the scene elements to create a final depth map c) allowing to create the stereoscopic pair.*

5.2.2.1 Using Only Three Cameras: The Tri-Rig

For a generic scene containing many elements, it becomes unreasonable to use as many cameras as depth constraints. But a scene with a low number of constraints can be acquired with a small number of cameras. In our simplified scene layout with two elements, a constraint for the depth and roundness of the subject of interest, and a depth constraint on the background, result in a three camera configuration that we call the "Tri-Rig". The three cameras are placed as follows.

We choose the camera used by the director to establish the 2D frame as the left most camera. Then, the second camera is chosen so that the subject of interest is perceived at the desired depth ($z'(z_s) = z'_s$), and with the desired roundness factor ($\rho(z_s) = \rho_s$). For example, if the subject's depth is set at the depth of the screen and the desired roundness is $\rho_s = 1$, the obtained baseline between the first and the second camera is

$$b_{\text{round}} = b' \frac{H}{H'}. \tag{5.24}$$

This baseline also establishes the convergence window width and distance (W, H).

The third camera is placed so that its baseline with the first camera acquires the background as required by the director, i.e. the perceived depth of the background is the desired one. We name this baseline b_{div}, as it is usually chosen to avoid the ocular divergence arising with the roundness baseline b_{round}. The convergence

window width and distance (W, H) are chosen to be equal to the configuration acquiring the subject. For example, if the director wants the background elements at infinity to be perceived at infinity in the projection room, the obtained baseline corresponds to the *canonical setup*:

$$b_{\text{div}} = b' \frac{W}{W'}. \tag{5.25}$$

5.2.2.2 Generating the Final Stereoscopic Pair

Now that we have acquired the three (or more) images, we need to compose them into a stereoscopic pair. The left most image created by the director is used as the left image of the stereoscopic pair. Then we need to render the right image corresponding to the director's constraints: our target view.

The right image has to be rendered as the composition of the other images. In the "Being on the field" approach, we had virtual cameras establishing the positions and parameters of the cameras to render, but in this case we do not know where the camera to be rendered is. Moreover, the image we want to render can not be obtained with a standard pinhole camera, as the different constraints on the scene elements result in different camera positions. Let us analyze the image formation process of the desired right image.

We can describe the image formation process of the target view with the composition of two functions. First, a function $d(z)$ transforms scene depth values z into disparity values d, and then a *disparity mapping* function $\phi(d)$ (see Sec. 3.4.1) transforms the image, so that an acquired disparity d is mapped into the desired disparity $\phi(d)$. Let us recall that $\phi(d)$ is generally assumed to be increasing monotonic, to avoid mapping farther objects of the scene in front of nearer objects of the scene. The depth to disparity function can be defined with any director's constraints establishing an initial acquisition setting. We name this acquisition setting the *reference* acquisition setup (b_r, H_r, W_r). It establishes the depth to disparity mapping (see Sec. 3.1.6):

$$d(z) = \frac{b_r}{W_r}\left(1 - \frac{H_r}{z}\right). \tag{5.26}$$

Then, each supplementary constrain is taken into account by defining a *control point* on the $\phi(d)$ function. A depth constraint $z'(z_e) = z'_e$ constrains $\phi(d)$, whereas a roundness constraint $\rho(z_e) = \rho_e$ also constrains $\phi(d)$ and $\phi'(d)$ (see Sec. 3.4.1.4). Note that a depth constraint plus a fixed convergence distance, fully constrain the acquisition system. Thus, $\rho(z_e) = \rho_e$ is also defined, which establishes a constraint on $\phi'(d_e)$. The constraint on the convergence distance and the following constraint on $\phi'(d_e)$ is usually imposed on the background elements of the scene. Once all *control points* have been set, the final continuous function $\phi(d)$ can be computed with any interpolation technique exactly interpolating the control points and its derivatives. To generate it we can use any linear (Pitié *et al.*, 2012) or non-linear operator (Lang *et al.*, 2010), or even cubic splines. Transitions between

operators should be smooth, so that $\phi'(d)$ is properly defined for all disparity values. Otherwise the roundness factor would not be defined at depths where $\phi(d(z))$ is not differentiable. Of course the shape of the disparity mapping function varies depending on the *reference* acquisition setup. Let us illustrate two different $\phi(d)$ functions obtained with our "Tri-Rig" configuration.

If we choose the baseline b_{div} as the reference, the depth of the background is properly mapped to the desired disparity, but the roundness factor on the subject is not the desired one. The goal of the disparity mapping function is to add roundness at the subject of interest depth z_s, i.e. to expand the disparities around the subject disparity $d(z_s)$. The function $\phi(d)$ must of course preserve the constraints on the background, i.e. the values $\phi(d_b)$ and $\phi'(d_b)$. In Fig. 5.15 we illustrate the shape of the obtained $\phi(d)$ function.

Similarly, if we choose the b_{round} as the reference baseline, the depth and roundness factor on the subject are properly acquired, but the disparity corresponding to the background elements does not fulfill the background constraints. Elements at the background depth have too high disparity values. In this case, the goal of the disparity mapping function is to compress the disparities so that the background disparities are mapped to the desired ones, i.e. $\phi(d_b)$ and $\phi'(d_b)$. In Fig. 5.16 we illustrate the shape of the obtained $\phi(d)$.

Once the *reference* baseline is chosen and the disparity mapping defined, we can compute the warps from the input images into the target image.

5.2.2.3 Generic Warps with Disparity Mapping

The goal now is to establish the generic form of the backward warp map τ_i transforming a point $\boldsymbol{x} = (x, y, 1, z(x, y)^{-1})$ in the input image, into the point $\boldsymbol{u}' = (u', v', 1, d')$ in the final image. We assume that we have a geometric proxy, the camera matrices and the disparity mapping function. Thus we can establish the correspondences between the input image and the target image.

We start by computing the 3D scene point $\boldsymbol{p}$ associated with $\boldsymbol{x}$. As we did in Sec. 4.5.2.1, we use the inverse of the reconstruction matrix $\tilde{\boldsymbol{P}}_i^{-1}$ of the camera i relating both points: $\boldsymbol{p} = \tilde{\boldsymbol{P}}_i^{-1}\boldsymbol{x}$. Then, we project $\boldsymbol{p}$ into the *reference* camera point $\boldsymbol{u} = (u, v, 1, d)$ with the reconstruction matrix of the reference camera $\tilde{\boldsymbol{P}}_r$. Let us recall that in Sec. 4.5.2.1 we did not have a reference baseline and so we used the reconstruction matrix associated to the *normalized disparity* mapping. In this case we do have a reference baseline and the mapping between the depth of a scene point and the disparity is given by Eq. 5.26. To obtain the reconstruction matrix $\tilde{\boldsymbol{P}}_r$ we simply replace f, b, H, W with b_r, H_r, W_r in Eq. 3.25 and obtain

$$\tilde{\boldsymbol{P}}_r = \begin{pmatrix} \frac{H_r}{W_r} & 0 & 0 & 0 \\ 0 & \frac{H_r}{W_r} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{b_r}{W_r} & -b_r\frac{H_r}{W_r} \end{pmatrix}. \tag{5.27}$$

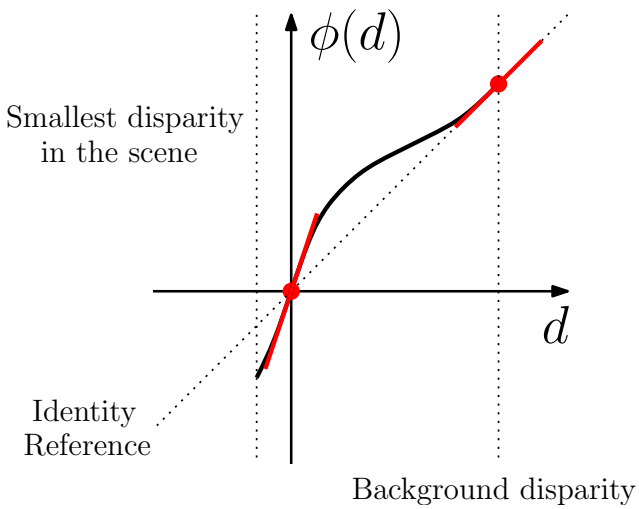


Fig. 5.15: *Disparity mapping function $\phi(d)$ expanding the disparity range near the zero disparity, i.e. the subject of interest depth. Disparity values after the expansion are compressed and the disparity values of the background are preserved.*

The last step is to use the disparity mapping function $\phi(d) = d'$ to obtain the desired point. We note as $\boldsymbol{\phi}$ (the vectorial version of ϕ) the function displacing the x-coordinate of the projected point $\boldsymbol{u}$ in the image. This transformation can be written as

$$\boldsymbol{\phi}(\boldsymbol{u}) = (u + \phi(d) - d, v, 1, \phi(d)). \tag{5.28}$$

The backward warp map τ_i transferring a point $\boldsymbol{x}$ in the image i into $\boldsymbol{u}'$ in the final view can be obtained as the composition of $\tilde{\boldsymbol{P}}_i^{-1}, \tilde{\boldsymbol{P}}_r$ and $\boldsymbol{\phi}$. As we did in Sec. 4.5.2.1, we write the left product of $\tilde{\boldsymbol{P}}_i^{-1}$ with $\tilde{\boldsymbol{P}}_r$ as

$$\tilde{\boldsymbol{T}}_i = \tilde{\boldsymbol{P}}_i^{-1}\ \tilde{\boldsymbol{P}}_r, \tag{5.29}$$

and obtain

$$\tau_i = \boldsymbol{\phi} \circ \tilde{\boldsymbol{T}}_i. \tag{5.30}$$

The Forward Warp Map To compute the forward warp map $\beta_i = \tau_i^{-1}$ we need $\boldsymbol{\phi}(\boldsymbol{u})$, i.e. $\phi(d)$, to be invertible. Note that Lang *et al.* (2010) allowed multiple depth values to be mapped to exactly the same target depth. By doing so the forward warp map β_i is not properly defined at this target depth. However, as disparity values are in $\mathbb{R}$, it is easy to impose a strict monotonic condition on ϕ. In this same work, Lang *et al.* (2010) propose to use the non-linear operator $\phi(d) = log(1 + sd)$, with $s \in \mathbb{R}$, to compress the disparity range, which is strictly increasing. Thus the forward warp map β_i is properly defined.

If $\phi(d)^{-1}$ is defined, then $\boldsymbol{\phi}(d)^{-1}$ is also defined and the forward warp map β_i of a point in $\boldsymbol{u}'$ is given by

$$\tilde{\boldsymbol{x}} = \tilde{\boldsymbol{T}}_i^{-1} \boldsymbol{\phi}^{-1}(\boldsymbol{u}'), \tag{5.31}$$

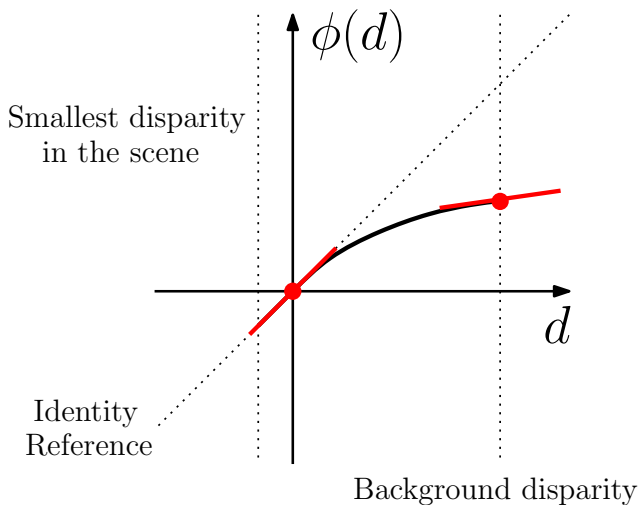


Fig. 5.16: *Disparity mapping function $\phi(d)$ compressing the disparity range for the background values. Disparity values at the convergence depth are preserved.*

where $\boldsymbol{\phi}^{-1}(\boldsymbol{u}')$ is a 4-dimensional row vector, and $\tilde{\boldsymbol{T}}_i^{-1}$ a 4×4 matrix. To obtain the final coordinates in the input image we need to normalize $\tilde{\boldsymbol{x}}$ with the third coordinate.

The Weights With the expressions of τ_i and β_i we can compute the terms $|\det D\beta_i|$ (and $|\det D\beta_i|''$) as well as $\frac{\partial \tau_i}{\partial z}$. The warps τ_i and β_i are the composition of the expressions obtained in Sec. 4.5.2.1 with $\boldsymbol{\phi}$. Thus to compute the derivatives, we only have to apply the chain rule. The spatial derivatives from $\tilde{\boldsymbol{T}}_i$ can be computed with the equations obtained in Sec. 4.5.2.3, and the derivatives with respect to depth can be computed with the equations obtained in Sec. 4.5.2.2. Then we only need to compute $\frac{\partial \boldsymbol{\phi}}{\partial z}$ and $|\det D\boldsymbol{\phi}|$. To compute $\frac{\partial \boldsymbol{\phi}}{\partial z}$ we use the relation between depth and disparity given by Eq. 5.26 and to compute $|\det D\boldsymbol{\phi}|$ we use that $\phi(d)$ is independent of the x and the y-coordinates. Thus we have all magnitudes needed to compute the final images.

Let us point out, that the weights of the method proposed by Wanner and Goldluecke (2012) can be computed as they rely on $|\det D\beta_i|$. However, at this moment it is unclear how to compute the weights for the method proposed by Buehler *et al.* (2001), as they rely on angles between optical rays which could be affected by the $\boldsymbol{\phi}$ function.

5.2.2.4 The World Distortion

The actual implementation of the forward and backward warps needs special attention, as the occlusion handling in this process is delicate. In the backward warp map τ_i the position of the projected element $\boldsymbol{u}$ is modified by $\boldsymbol{\phi}$. Two geometric elements $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ projected at two different image locations $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$, may have the same image coordinates after the disparity mapping warp. The element

with a lower disparity value occludes the element with a higher disparity value. The visibility test, also known as z-buffering, should use the final disparity mapped values d' instead the depth of the element to the camera z.

Similarly, when computing the forward warp map β_i, two image points $\boldsymbol{u}'_1$ and $\boldsymbol{u}'_2$, may be warped into points $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ having the same x- and y-coordinates. Thus, to first warp the image and then apply $\tilde{\boldsymbol{T}}_i^{-1}$ can not be done by storing the warped values in a classical single planar buffer. Some elements of the buffer will be empty, whereas other elements will have multiple assignments. A special pipeline should be implemented.

The occlusion handling in the render engines such as OpenGL (Woo *et al.*, 1999) has been optimized over the years for standard pinhole camera projections. Ideally, we would like to take advantage of the actual rendering techniques. Hence we propose not to apply the disparity mapping $\phi(d)$ in the images, but to distort the world accordingly with a function $\boldsymbol{\Phi} : \mathbb{R}^3 \to \mathbb{R}^3$ before the pinhole camera projection. In Fig. 5.17 we illustrate a scheme of how $\boldsymbol{\Phi}$ transforms a 3D point $\boldsymbol{p}$ into $\boldsymbol{p}'$. With this pipeline we can compute the desired image warps with a classic depth occlusion handling, which is done by the rendering engine.

To compute $\boldsymbol{\Phi}$ we first define $\Phi : \mathbb{R} \to \mathbb{R}$ which acts on the depth of a point. The function $z' = \Phi(z)$ is directly related to $d' = \phi(d)$ by the relation between depth and disparity. We first convert the depth z into the disparity d, and then we disparity map it into d'. To obtain the z' value we reconstruct the mapped disparity. Both depth to disparity and disparity to depth mappings are established by the reference baseline, i.e. Eq. 5.26 and its inverse. Then, given a 3D point, if we transform its depth with Φ, and project it, the final disparity of the projected point is d'. To completely define $\boldsymbol{\Phi}$, we still have to define its x and y transformation to obtain a coherent 3D distortion of the world. As the frame of the left camera is chosen by the director, a natural constraint on $\boldsymbol{\Phi}(\boldsymbol{p})$ is not to modify the projection of the geometry into the left camera. This way, the image of the left camera is unaffected by the geometry distortion. With this constraint together with the depth mapping function $\Phi(z)$, $\boldsymbol{\Phi}(\boldsymbol{p}) = \boldsymbol{p}'$ is fully defined. The distorted 3D scene point $\boldsymbol{p}'$ lies on the viewing ray $\overline{\boldsymbol{p}\boldsymbol{c}_l}$, where $\boldsymbol{c}_l$ is the optical center of the left camera, and the amount

$$\begin{tikzcd}
\mathbf{p} \arrow[r, "\Phi(\mathbf{p})"] \arrow[d, "\tilde{\mathbf{P}}_r(\mathbf{p})"'] & \mathbf{p}' \arrow[d, "\tilde{\mathbf{P}}_r(\mathbf{p}')"] \\
\mathbf{u} \arrow[r, "\phi(\mathbf{u})"'] & \mathbf{u}'
\end{tikzcd}$$

Fig. 5.17: *The definition of $\Phi(\boldsymbol{p})$ as the equivalent world distortion creating the image warp defined by $\boldsymbol{\phi}(\boldsymbol{u})$: $\boldsymbol{\phi} \circ \tilde{\boldsymbol{P}}_r = \tilde{\boldsymbol{P}}_r \circ \Phi$.*

by which the point $\boldsymbol{p}$ is displaced along the viewing ray is determined by $\Phi(z)$. In Fig. 5.18 we illustrate a scheme of the proposed world distortion. In practice $\boldsymbol{\Phi}(\boldsymbol{x})$ can be efficiently implemented with a vertex shader.

With the proposed world distortion, angles between the desired optical ray and the input camera ray can be computed as illustrated in Fig. 5.19. Given a point $\boldsymbol{u}'$ on the reference image, its 3D point $\boldsymbol{p}'$ in the distorted world can be computed. Then the desired viewing ray is defined in the distorted world. This ray can be undistorted with $D\boldsymbol{\Phi}^{-1}$ evaluated at the depth of $\boldsymbol{p}'$. The undistorted ray can now be compared to the ray between $\boldsymbol{p}$ and $\boldsymbol{c}_i$, where $\boldsymbol{p}$ is the undistorted version of $\boldsymbol{p}'$, and $\boldsymbol{c}_i$ the optical center of the input view. The weights of the method proposed by Buehler *et al.* (2001) can now be computed.

5.2.2.5 Visibility

By construction, the left view of the proposed "Tri-Rig" is the raw output from the camera. No occlusion, other than an *intruder-scene*, may arise. However, occlusions in the right final rendered view can arise. In Fig. 5.20 we illustrate how the subject may occlude a region of the background needed in the final image.

As we saw in Sec. 5.7, if the subject or the background are not convex, the same scheme from Fig. 5.20 applies for self-occlusions. If b_{div} is chosen as the reference baseline, then the world distortion modifies the subject of interest and the world distortion may introduce self-occlusion regions. Similarly, if b_{round} is chosen as the reference baseline, and the background is not convex, self-occlusion areas may arise.

Although the needed texture may be acquired by another image of the "Tri-Rig", it is highly probable that some region of the final image is not visible by any other view. As we discussed in Sec. 5.1.4, if the texture is not recovered by any of the "Tri-Rig" cameras, a possible solution could be to use auxiliary cameras to acquire the missing texture, or if no other cameras are available, then use an inpainting method to fill the missing regions in the image.

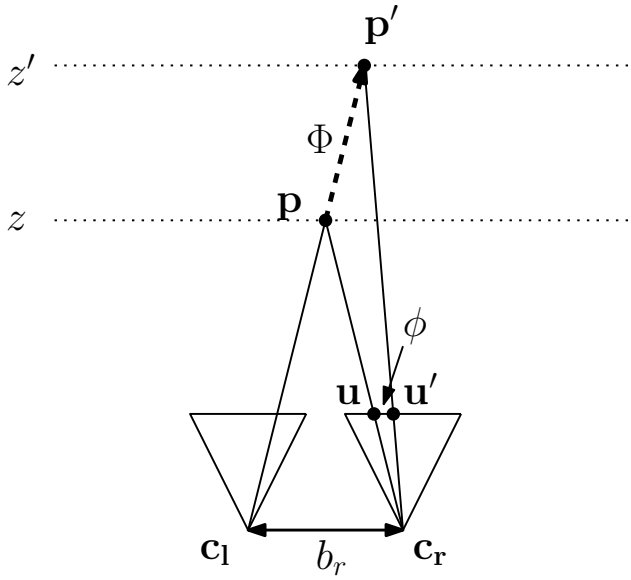


Fig. 5.18: *The proposed world distortion $\Phi(\boldsymbol{p})$ as the composition of $d(z)$, $\phi(d)$ and $z'(d')$.*

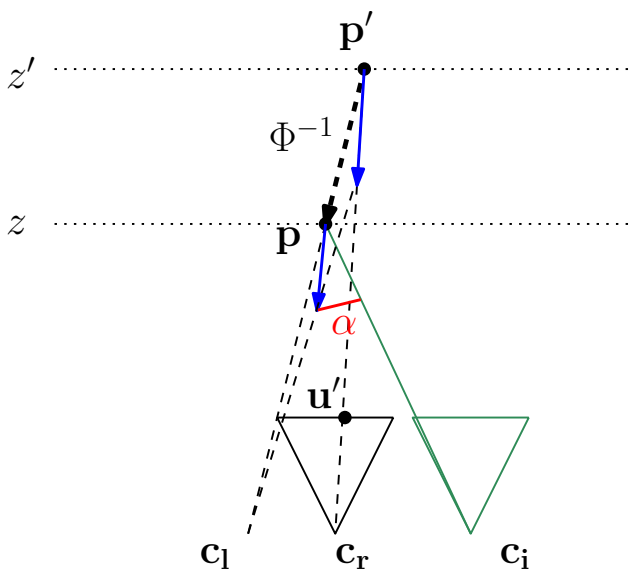


Fig. 5.19: *The Jacobian of the inverse of the proposed world distortion $\boldsymbol{\Phi}^{-1}$ allows to undistort viewing directions into the 3D world. Angles between optical rays from the distorted world and the 3D world can be computed. The viewing ray from the image point $\boldsymbol{u}'$ intersects with the distorted word at $\boldsymbol{p}'$. The direction of the viewing ray can be undistorted, and then compared to the viewing ray from the camera i in the 3D world. The angle between both rays is α.*

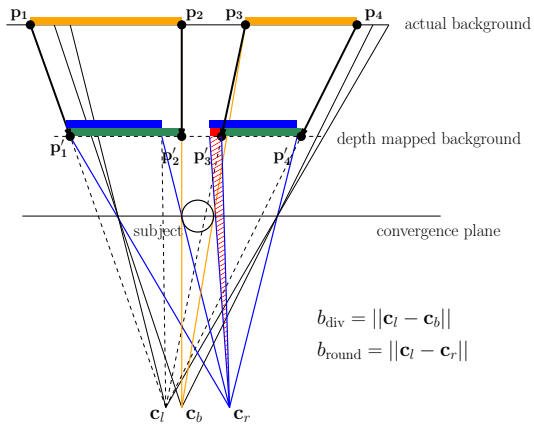
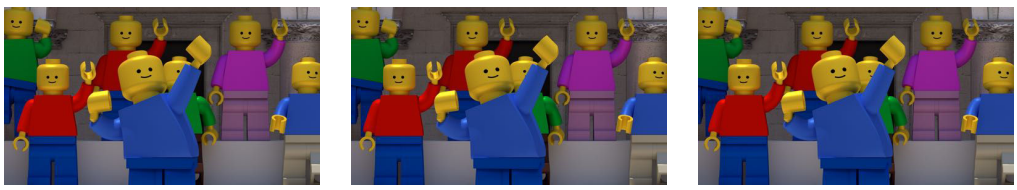


Fig. 5.20: *Example of subject-background occlusion. The roundness baseline is used as reference baseline. The depth mapping reduces the depth of the background to decrease its disparity. A region needed by the reference camera (red rectangle) is not acquired by the background camera. The background camera at* $\mathbf{c}_b$ *acquires the region of the background defined by the points* $\mathbf{p}_i$ *(orange rectangles). The points* $\mathbf{p}_i$ *are mapped into* $\mathbf{p}'_i$ *as well as the visible area (green rectangles). The needed area of the background to render the final image is shown in blue. The red rectangle indicates the region not acquired by the background camera.*

5.2.3 Proof of Concept

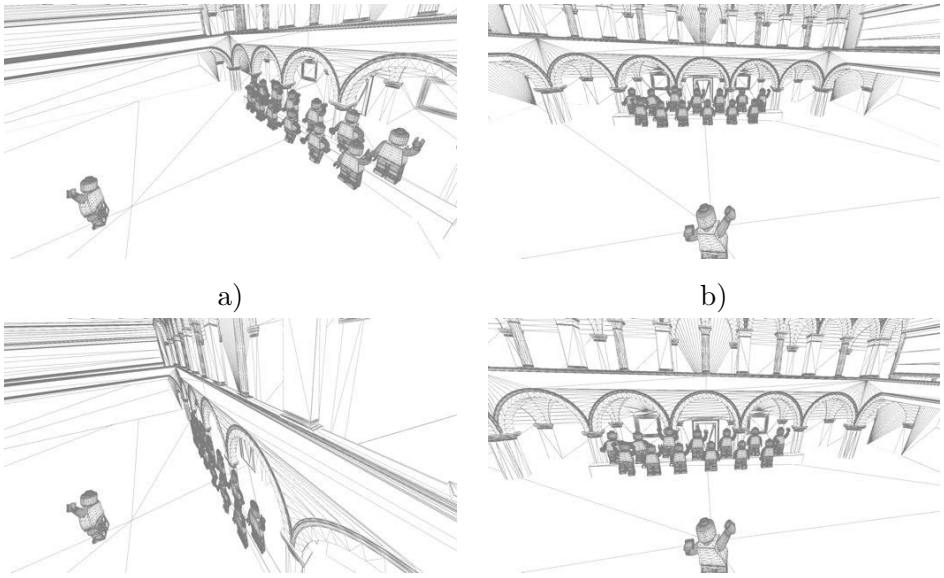
In this section we present a proof of concept of the proposed approach. We created a synthetic dataset with our simplified layout, where the ground truth geometry is available. In Fig. 5.21 we illustrate the three rendered images of the dataset. We used the baseline acquiring the subject of interest with the desired roundness factor as the reference baseline. Thus the disparity mapping function $\phi(d)$ had the shape illustrated in Fig. 5.16. In Fig. 5.22 we show the world distortion created by the corresponding $\mathbf{\Phi}(\boldsymbol{x})$. The depth of the subject of interest is preserved, while the background elements are pulled forward to decrease their disparity values on the final image.

In Fig. 5.23 we show the final rendered images with the methods proposed by Buehler *et al.* (2001), Wanner and Goldluecke (2012), as well as our IBR method. Visually, the rendered images do not present any noticeable difference. As the target image does not correspond to any pinhole camera, we do not have a reference image to compare with, and thus we can not numerically evaluate the obtained results.



a) b) c)

Fig. 5.21: *The "tri-rig blender lego" dataset: a) and b) are the left and right images acquiring the background. a) and c) are the left and right images acquiring the subject of interest.*



a) b) c) d)

Fig. 5.22: *The world distortion for the "tri-rig blender lego" dataset: a) and b) are views of the 3D scene c) and d) are the same views of the distorted 3D scene. The background depth is compressed, so that further elements create a smaller disparity in the final rendered image.*

We discuss how our results could be evaluated in the next Section.

In Fig. 5.24 we show closeups of the regions in the rendered images, which are not acquired by any source image. These regions are very small compared to the large black areas obtained with the "Quadri-Rig" approach shown in Fig. 5.11.

5.2.3.1 Future Evaluation

We believe that a subjective evaluation of the obtained results should be conducted in the future to assess the proposed approach. Of course the final quality of the rendered images is strongly dependent on the geometric reconstruction, thus two different experiments should be conducted to assess two different questions.

The first set of experiments should assess the validity of the proposed camera model. Is the "Tri-Rig" capable to create compelling stereoscopic images with the desired roundness factor on the subject and avoiding ocular divergence? The groundtruth geometry should be used to generate the final images and avoid visual artifacts created by a poor 3D reconstruction. The rendered images could be compared to disparity mapping methods using two images such as Lang *et al.* (2010), Yan *et al.* (2013) or Devernay and Duchêne (2010), which should of course also benefit from the goundtruth reconstruction. Although we believe that the generated images would be considered to contain less artifacts, specially in regions where important occlusions arise, without the subjective experience we do not have yet any proof.

The second set of experiments should be conducted with rendered images relying

Fig. 5.23: *First column: the right final image, rendered with the different methods. First row is rendered with Buehler* et al. *(2001) (ULR), second row is rendered with Wanner and Goldluecke (2012), and the third row is rendered with our approach. Second column: difference of the image with ULR. No noticeable difference is visible between the rendered images by the 3 methods.*

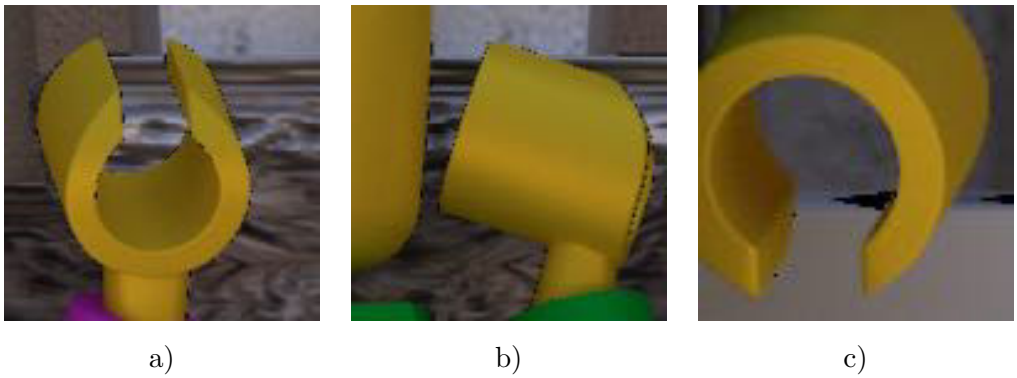


Fig. 5.24: *Occluded regions in the "Tri-Rig" configuration. Very few pixels around depth discontinuities are not acquired by any camera. a) and b) A few black occluded pixels are visible around the hand of the lego models. c) The black triangular region in the center of the closeup is not acquired by any camera.*

on computed depth values from the input images. The rendered images could be compared to the disparity mapping state of the art techniques cited above. Thus the actual implementation of the model would be evaluated.

An interesting comparison could be done with the stereoscopic images created with two actual cameras using either the non-diverging baseline, or the roundness factor baseline. As the acquired images do not contain any visual artifacts, users should choose between the "cardboard effect", the ocular divergence or the proposed images containing potentially visual artifacts.

In addition, it could be interesting to see how no-reference stereoscopic measures, e.g. Akhter *et al.* (2010); Chen *et al.* (2013), correlate to the subjective evaluation. While those methods can detect artifacts in the images and would be useful for the second set of experiments, they do not account for "cardboard effect" or ocular divergence in the images. This is why we belief that a user study should be conducted.

5.2.4 Tri-Rig Discussion

We presented the "Tri-Rig", a camera model based on three cameras to create stereoscopic images with perspective distortions of the acquired scene. To conclude the "Tri-Rig" section we discuss how to handle inconsistent constraints, as well as a followup question raised by the proposed world distortion.

5.2.4.1 Inconsistent Constraints

At the time of the constraints specification, the director may be unaware of the actual depth of the scene elements, and some constraints may be incompatible with each other. For example, two elements at the same actual depth may be constrained to be at different depths in the final images. One way to address this problem is to apply a different $\phi(d)$ and $\Phi(z)$ depending on the location of the element in the scene $\boldsymbol{x}_e$, i.e. $\phi(\boldsymbol{x}_e)$ and $\Phi(\boldsymbol{x}_e)$. Transition areas between the different regions should be handled carefully. On one hand, inconsistent monoscopic depth cues could be easily in conflict, e.g. interposition or perspective. On the other hand, if the disparity mapping function ϕ depends on the spatial coordinates, the computed weights described in Sec. 5.2.2.3 require that the derivatives $\frac{\partial \phi}{\partial x}$ and $\frac{\partial \phi}{\partial y}$ are properly defined. Thus ϕ should be defined as a continuous, differentiable function.

5.2.4.2 IBR in Distorted Worlds

The multi-rig problem is known to be an IBR problem in a world were the optical rays are not straight, as pointed out by Pinskiy *et al.* (2013). The IBR problem in a world where the rays are not straight is, according to McMillan and Bishop (1995), still a plenoptic sampling problem. The question reduces to identify which ray needs to be reconstructed and how to reconstruct it from the available sampled rays. However, a question arises: should the same *desirable properties* from Buehler

et al. (2001) prevail in a world were the optical rays are not straight? Moreover, if the answer is "no", which one of them and how should they be adapted?

In our approach we distorted the world to obtain straight viewing rays. This way the occlusions are properly handled. However, the distorted world should not be used to compute any metric values, such as angles or distances. Rendering techniques such as illumination relying on metric values should not be computed in the distorted world as they would lead erroneous results.

5.3 Discussion and Conclusion

We presented two different approaches to create stereoscopic images with long focal lengths, each one of them guided by a different artistic intention of the director. The first one allows the director to "get closer" to the scene while the second one allows the director to create perspective distortions of the world. Each intention has allowed us to deduce the corresponding camera model acquiring the needed images. Let us compare the obtained camera models and discuss future lines of investigation.

5.3.1 Tri-Rig vs. Quadri-Rig

Although the target images have a very different goal, in the sense of stereoscopic mise-en-scene, we compare both camera models to highlight their advantages and flaws.

One of the attractive features of the "Tri-Rig" with respect to the "Quadri-Rig" is that one of the images, i.e. the left one, is acquired by a source camera. This is an important advantage, as the perceived quality of a stereoscopic pair of images is close (and sometimes equal) to the quality of the best of both images (Seuntiens *et al.*, 2006). Thus having the raw output of the camera as the left view provides the highest quality possible. In the "Quadri-Rig", both images are rendered, and thus may contain visual artifacts.

Another advantage of the "Tri-Rig" with respect to the "Quadri-Rig" is the visibility of the scene. As we illustrated in Fig. 5.7, large areas needed by the target images of the "Quadri-Rig" may not be acquired by any of the four actual cameras. In the "Tri-Rig" setup, only the right image may have image regions which are not recovered by any camera. Moreover, as we illustrated in Fig. 5.20, these regions are smaller than the regions shown in Fig. 5.7.

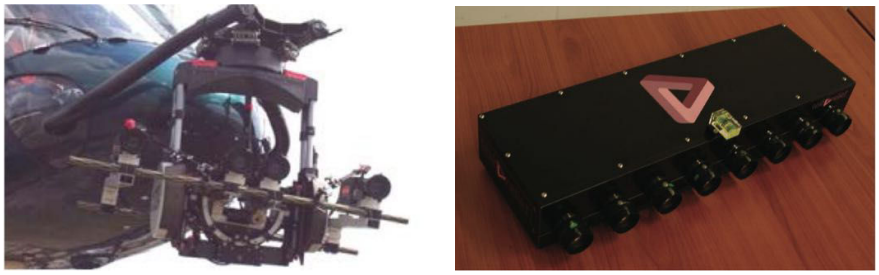
Although in this chapter we specialized the camera models to a scene with a simple layout, composed of a subject of interest and a background, the proposed camera models can be extended to scenes with more elements. For each new element of the scene, the "Quadri-Rig" needs to be extended with 2 more cameras, whereas the "Tri-Rig" only needs one more. As the focal length for all elements is the same, the left camera of the new obtained camera setup acquiring the new element can be set in the same exact position as the first left camera. We can summarize that

given N elements of the scene, the "Quadri-Rig" has a complexity of $2N$, while the "Tri-Rig" has complexity $N + 1$.

Because of the advantages of the "Tri-Rig" with respect to the "Quadri-Rig", the director could also use the "Tri-Rig" with the intention to "get closer" to the scene. The director could benefit from the advantages of the "Tri-Rig" and accept the created perspective distortions to the scene. In this case, similarly to the use of a zoom in 2D, the perspective distortion would be a consequence, but not the intention.

5.3.2 Actual Implementation

The proposed camera models use either 3 or 4 cameras. Although the resulting acquisition setup would be heavy and cumbersome, an actual implementation is feasible in practice. Commercial stereoscopic rigs proposed for example by Binocle (2015) and 3ality Technica (2015), allow to change the baseline between the cameras as well as the focal length of the cameras. Motion control units allow to precisely and synchronously control their focal length parameters as well as their positions. Moreover, some multiview acquisition prototypes have been created for special shootings, like for example the 4-camera rig used by Binocle to shoot "La France entre ciel et mer" (France between sky and sea), or an 8-camera rig specially designed to acquire images to be displayed on autostereoscopic devices (Prevoteau *et al.*, 2010). In Fig. 5.25 we reproduce images of these multiview prototypes, which could be the starting point of an actual implementation of the proposed camera models.



a) b)

Fig. 5.25: *a) 4 camera rig used by Binocle in the shooting of "La France entre ciel et mer" (France between sky and sea). b) 8 camera rig proposed by Prevoteau* et al. *(2010).*

5.3.3 Autonomous Calibration and Depth Computation

Another advantage of the proposed system is that the acquired images have an important overlap. This could allow to calibrate each camera with respect to the others with an offline method. The calibration data could be then embedded in the motion control system driving the rig and retrieved in real-time.

In our proof of concept, the 3D reconstruction obtained with the pipeline described in Sec. 4.5.1 provided poor results. Even though we used auxiliary

cameras to help in the reconstruction process, the obtained results created important visual artifacts shown in Fig. 5.13. One way to address this issue is to aim at a specific reconstruction algorithm, specially tailored for the proposed camera configurations. For example, for the "Quadri-Rig", as each pair of cameras has the same focal length, we could use stereo methods to compute two pairs of disparity maps. Then both pairs of disparity maps could be further refined by combining the overlap region in both pairs. For the "Tri-Rig", as all cameras have the same focal, disparity maps between each pair of images could be computed. Then the trifocal tensor could be used to reject incoherent values and refine the final result. This way, the depth of the scene could be significantly improved and we could hope that most of the visual artifacts could be eliminated.

5.3.4 Future Evaluation

Future work should address the validation of the "Tri-Rig" generated images. As we do not have a reference image to compare with, we were not capable to assess the pertinence of the proposed camera model as well as the quality of the rendered images with its actual implementation. It is our belief that a subjective user study should be conducted to evaluate the proposed camera model.

6 Conclusion

6.1 Summary

In this thesis we studied how to create stereoscopic images with cameras using a long focal length and contributed two camera models to the domain of "3D cinematography" (Ronfard and Taubin, 2007, 2010). Each camera model follows a different intention of the director to create a different stereoscopic effect. The first one, the "Quadri-Rig", is driven by the intention to "get closer" to the scene. The second one, the "Tri-Rig", is initially driven by the intention to create perspective distortions of the acquired scene, although it can also be used to "get closer" to the scene.

We contributed a new visualization tool, the "virtual projection room", allowing to better understand the complex transformation between the acquired 3D scene and the 3D scene perceived by the spectator in the projection room. With the proposed tool, we illustrated the geometric distortions arising with different acquisition and projection settings, e.g. the "cardboard effect". We then analyzed how a change in the projection configuration could lead to important distortions. We reviewed the state of the art techniques that address the problem of how to adapt content created for a target projection configuration into a different projection configuration. These techniques introduced the concept of *disparity mapping*, which is a clever function targeting to reduce the distortions of the perceived depth. We studied how the mathematical formalization of the constraints are affected by the *disparity mapping function*, and revisited the obtained acquisition configurations. We furthermore analyzed the geometric distortions arising when using acquisition cameras with long focal lengths. We saw the "cardboard effect", related to the roundness factor of the perceived depth, as well as the ocular divergence, a major cause of visual fatigue. We explained how the limitations of the existing state of the art methods prevent them to obtain the desired results. In order to overcome the limitations of the state of the art, we contributed two approaches to create stereoscopic images with long focal lengths.

In chapter 4 we contributed a new IBR generative model capable of explaining

most currently accepted intuitions of the state of the art in IBR (Buehler *et al.*, 2001), while retaining the advantage of the intrinsically parameter-free energies arising from the Bayesian formalism (Wanner and Goldluecke, 2012). The key theoretical contribution of the proposed method is the systematic modeling of the error introduced in the Lambertian image formation process via the inaccuracy in the estimates of the geometric proxy. We call this inaccuracy *depth uncertainty*, referring to the depth estimates from the input images. We extensively analyzed the theoretical implications of the obtained energy, discussing the formal deduction of the state of the art heuristics from our model. This work contributes the first Bayesian formulation explicitly deriving the heuristics of Buehler *et al.* (2001). From a practical point of view, we numerically evaluated the performance of our method for two cases. First we addressed a simplified camera configuration where all viewpoints are in a common plane, which is parallel to all image planes. This configuration is known as the Lumigraph (Gortler *et al.*, 1996). For this configuration we compared our results to the best existing method within the Bayesian framework (Wanner and Goldluecke, 2012). In a second set of experiments we dealt with the generic, unstructured configuration as proposed in Buehler *et al.* (2001). For this configuration we implemented the generic extension of Wanner and Goldluecke (2012) as well as the method proposed by Buehler *et al.* (2001), and compared our results to both of them. Experimental results showed that we achieve state of the art results with regard to objective measures on public datasets. Moreover, we are also capable of addressing super-resolution, capitalizing on the general framework established in Wanner and Goldluecke (2012). We also described the main limitation of the proposed approach, which is the dependency of the energy on the latent image u, and provided some hints on how future work can address this limitation.

Finally, we used our generative IBR model to deduce two camera models, which acquire the images allowing to create the desired stereoscopic effect. We detailed two different approaches to the stereoscopic mise-en-scene, leading to two different stereoscopic intentions, and contributed two different camera models. Although the proposed theoretical models can be applied to any generic scene, a complex scene with many elements needs in theory to be acquired by a high number of cameras. The resulting acquisition setup may result in practice in a complex and cumbersome device. Hence, we focused on a simplified layout scene consisting of a subject of interest and a background and deduced the corresponding camera models. The first one, the "Quadri-Rig", allowed to create images following the directors intention to "get closer" to the scene. The second model, the "Tri-Rig", allowed to create perspective deformations of the acquired 3D scene to generate the desired stereoscopic image by the means of the *disparity mapping* function. Because of the advantages of the "Tri-Rig" with respect to the "Quadri-Rig", the director could also use the "Tri-Rig" with the intention to "get closer" to the scene, by accepting that some perspective distortions would be added to the perceived scene. We saw that for the simplified layout scene with a subject of interest and a background, an actual implementation of both camera models can be implemented in practice. We generated datasets which allowed us to experiment with the camera models.

We observed two limitations in our approach, the occluded regions in the rendered images and the dependency on a good geometric proxy. The occluded regions in the rendered images are formed by the pixels which are not acquired by any actual camera. Thus, we have no information to render these regions. While in the images generated with the "Tri-Rig" the occluded regions are small, in the images generated with the "Quadri-Rig" the occluded regions may contain a large number of pixels. The second limitation is the dependency of the final quality of the rendered images on a good geometric proxy. As we rely on a small number of cameras, if the quality of the geometric proxy is poor, important visual artifacts appear in the final images. To conclude, we discussed how future work can address this issue.

6.2 Future Work

To conclude the manuscript we provide directions on how to improve our three main contributions, by improving the virtual projection room, the generative model, as well as the camera models. We also give a hint on how methods taking our results as an input could benefit from our computations to deduce an image uncertainty.

6.2.1 Improving the Virtual Projection Room

One of the contributions of the present work is the "virtual projection room" presented in Sec. 3.3. The visualization of the perceived depth by the spectator in the virtual projection room is purely based on the perceived depth from stereopsis. While the prediction of depth from stereopsis is in most cases accurate (Held and Banks, 2008), future work should address how monoscopic depth cues, conflicting or inconsistent with stereoscopic depth cues, may bias the perceived depth from stereopsis. For example, if we capture a baseball player throwing the ball into the camera, the spectator may perceive the ball going out of the screen because of the size increase in the image as the ball approaches the camera. Even if the depth from stereopsis indicates that the ball should be perceived at the screen depth, the audience may perceive it right in front of them. It would be interesting to study how monoscopic depth cues can be integrated into the "virtual projection room" for a more realistic perceived depth prediction. The human subjective factor and the high number of depth cues involved in the process, makes this line of investigation a very challenging and hard problem.

6.2.2 Improving the Generative Model

Our Bayesian approach to the IBR problem has a main limitation, which is the dependency of the energy on the latent image u. On one hand it makes the energy optimization process less straightforward, and on the other hand, the local computation of ∇u may generate visual artifacts. In Sec. 4.5.7 we studied how different possibilities could address the limitations. However, it remained unclear how they could be justified in a formal way. Thus, in our opinion, the research of better generative models should continue.

6.2.2.1 Better Error Modeling

As we saw in Sec. 4.5.1.3, the per pixel uncertainty may not follow a normal distribution, specially near an occlusion border where the measured depth values switch from front to back depth values. In our computation of the per-pixel uncertainty, we could easily verify if the observed depth distribution is normal, or not. This information could be helpful to improve the geometric error modeling in our approach. For example, one could try to model the error as a mixture of Gaussian distributions, which could represent the front and back depth of the occlusion border. Then, an input pixel could contribute its color to two different image locations in the target image with a moderate weight, instead to only contribute to one image location with a low weight due to the high depth uncertainty.

6.2.2.2 Include the non-Lambertian Assumption

Another obvious lead for improvement would be to extend the generative model to non-Lambertian scenes. Because the real world is non-Lambertian, it seems crucial to include this assumption to increase the generality of the generative model. However, in practice this may prove to be quite hard, as one would need to include general BRDF and lighting information to correctly model the transformation between input and novel views.

6.2.2.3 Work in the Gradient Domain

The *continuity* desirable property of Buehler *et al.* (2001), states that the contribution of a pixel at the boundary of the field-of-view should smoothly fall to zero. With the proposed generative model we could not find any evidence pointing in this direction. Instead to focus on how the contribution of an image could diminish along the visibility boundaries, future work could explore how the generative model could be extended to be applied to the gradient domain, and thus directly overcome the photometric inconsistencies between the images.

6.2.2.4 Time Coherence

In the present work we have not explicitly taken into account the time dimension, and thus our model handles each temporal frame independently from the others. The generative model could be extended to exploit the temporal coherence among a sequence of images. Of course, instead to rely on a 3D geometric proxy, we would need a 4D geometric proxy, including not only the 3D position of each scene element and its uncertainty, but also is predicted displacement. This way, images at different frames could contribute to the final view. Note that in the literature of time interpolation, usually the blending weights are computed with heuristics which still lack a formal deduction Lipski *et al.* (2010, 2014). A generative model taking into account the time dimension could try to answer if an input camera closer in

space and farther in time should be preferred over a camera farther in space but closer in time.

6.2.3 Improving the Camera Models

6.2.3.1 Specific 3D Reconstruction Methods

Even though we used auxiliary cameras to help in the reconstruction process, the 3D reconstruction obtained with *PMVS* (Furukawa and Ponce, 2010) and *Poisson Reconstruction* (Kazhdan *et al.*, 2006) did not provide a good 3D reconstruction. The images rendered with the "Quadri-Rig" using the computed 3D reconstruction yielded poor results with important visual artifacts.

One way to address this issue would be to aim at a specific reconstruction algorithm, specially tailored for the proposed camera configurations. For example, for the "Quadri-Rig", we could use stereo methods to compute two pairs of disparity maps. Then the disparity maps could be refined by combining the overlap region in both pairs. For the "Tri-Rig", as all cameras have the same focal length, disparity maps between each pair of images could be computed. Then the trifocal tensor could be used to reject incoherent values and refine the final result. This way, the depth of the scene could be significantly improved and we could hope that some of the visual artifacts could be eliminated.

6.2.3.2 Tri-Rig Subjective Evaluation

The rendered image by the "Tri-Rig" approach does not correspond to any pinhole camera. Thus, we did not have a reference image to compare with, and could not numerically evaluate the obtained results. Although it would be interesting to see how no-reference stereoscopic measures, e.g. Akhter *et al.* (2010); Chen *et al.* (2013), evaluate the rendered images, those measures neither account for the "cardboard effect" nor ocular divergence in the images. This is why we belief that a user study should be conducted.

A first set of experiments should asses the validity of the proposed camera model by using the groundtruth geometric proxy of the scene. The experiments should address the question: if the geometric proxy is perfect, do the desired images create the expected stereoscopic effect? Then, if the camera model is validated, a second set of experiments should asses the quality of the rendered images by an actual implementation. If the quality of the rendered images is poor, efforts should focus on how to obtain a better 3D reconstruction.

6.2.3.3 Geometric Reconstruction aware Camera Models

When we deduced the camera models, we assumed the elements of the scene to be punctual. It would be interesting to study how the obtained camera positions may vary depending on the actual geometry of the acquired element.

For example, an interesting application to that problem would be the camera placement computation to acquire a scripted live performance. For example, imagine a theater where a live performance takes place and the actual camera placement is constrained. The decors as well as the actors could be approximated by a geometric proxy, and, as the movements of the actors are scripted, their position in space could be approximated. A director acquiring the performance could then place the virtual cameras at any position, and with the proposed framework, the positions and focal lengths of the actual cameras could be determined in order to maximize the quality of the virtual shots.

6.2.4 Exploiting Image Uncertainty

While we propose to benefit from the uncertainty of the geometric proxy, we can offer an uncertainty measure to any image processing technique taking our result as an input, e.g. image inpainting or image compression algorithms. For each rendered pixel we could easily compute two values: the *sum of the weights* of all contributing cameras, and the *color variance* of the contributing views. The first value accounts for how many cameras contribute to the pixel and in which "quality", in terms of resolution and angular deviation. The second accounts for the color variation proposed by the input views. A high color variance arises in two cases. If the observed geometric element is Lambertian, then the geometrical proxy (or the camera calibration) must be wrong. The same 3D point projects onto pixels of the input image which have different colors. The second possibility is that the observed scene is not Lambertian, thus different cameras observing the geometry from different angles observe different colors. In contrast to a high color variance, a low color variance indicates that the observed scene has a Lambertian, coherent geometry. Thus pixels with a low color variance and a high *sum of weights* should be more likely to have the actual color of the scene.

This information could be useful, for example, to encoding algorithms. Specially those aiming explicitly to 3D video encoding (Matsuyama *et al.*, 2012) could benefit from the color variance and the sum of weights information. The parts of the image with a high sum and low variance are likely to have a high quality, the compression algorithm could decide to compress them less. The parts of the image with high color variance and low sum of weights are likely to include visual artifacts, the algorithm could choose to apply a more aggressive compression to these image regions.

A

Dynamic Stereoscopic Previz

In this appendix we briefly present the *Dynamic Stereoscopic Previz* (Pujades *et al.*, 2014).

A.1 DSP Presentation

The goal of the simulator is to provide a *Previsualization* environment to the artists, to "see their movie before they shoot it" (Proferes, 2008). The director and the stereographer face an important question: how to set the acquisition parameters e.g., baseline b, the focal length ($f \propto \frac{W}{H}$) and convergence distance H, in order to obtain the desired 3D effect in the projection room. As we saw, this choice is difficult because the relationship between the 3D of the acquired scene and the perceived 3D in the projection room is complex. In recent years, expert directors, stereographers and researchers have proposed useful rules of thumb to overcome these difficulties, by containing the acquisition parameters in a "3D safe zone". For example the *1/30th rule* states that *"the interaxial distance should be 1/30th of the distance from the camera to the first foreground object"* (Mendiburu, 2009). A more sophisticated technique was proposed by Oskam *et al.* (2011), with an automatic stereoscopic camera control, providing a safe experience while exploring a virtual world. While these rules are very handy for safe filming, they happen to be also very limiting in terms of 3D creativity. In order to create novel 3D narratives (Mendiburu, 2011), some"not-so-safe" configurations should also be explored. Two main drawbacks make this exploration difficult. The first is that an actual exploration of stereoscopic configurations involves expensive equipment and time-consuming experiences. The second is that the director can not often *see* while shooting how the 3D shot will look like in the final projection room. Because the perceived depth depends on the size of the screen, one would ideally need a monitoring screen of the size of the target screen. The *Dynamic Stereoscopic Previz* (DSP) is a video game where the goal is to shoot a stereoscopic film. The user first models and animates a 3D scene using Blender (2015). Then the user places a stereoscopic rig in the scene and

adjust the shooting parameters at will (b, H, W). The user also sets the parameters of the *virtual projection room* (b', H', W'), and sees how the acquired images are perceived by the spectator. The virtual projection room is updated in real-time, as the user changes the shooting parameters.

A.2 DSP In Action

We tested our DSP tool during the shooting of a short stereoscopic movie. The short movie "Endless Night" takes place in an apartment, which we re-created (see Fig. A.1d). Based on the director's storyboards, we created previz animations for ten shots of the movie, one of which is presented in Fig. A.2. DSP takes as input an annotated storyboard (see Fig. A.1 a, b, c). For each shot the storyboard provides the first and the last frames, together with floor-plan view drawings of the desired camera and actors movements. The storyboard also contains written annotations on the desired stereoscopic mise-en-scene (shallow or deep shot, in front or at the back of the screen). In order to show the difficulty of the stereoscopic parameter setting, we present the panning shot, where the actress moves out of the bedroom and enters the living room. The shot is challenging because the motion of the camera introduces important changes on the acquired scene volume. The movement of the actress also gives multiple choices to set the convergence distance. Several stereoscopic choices are possible depending on the desired 3D effect. DSP allows to play with different stereoscopic configurations, by dynamically changing the baseline b and the convergence distance H. In Fig. A.2 we present the action recorded by DSP, as well as the actual rushes from a test shooting of the scene.

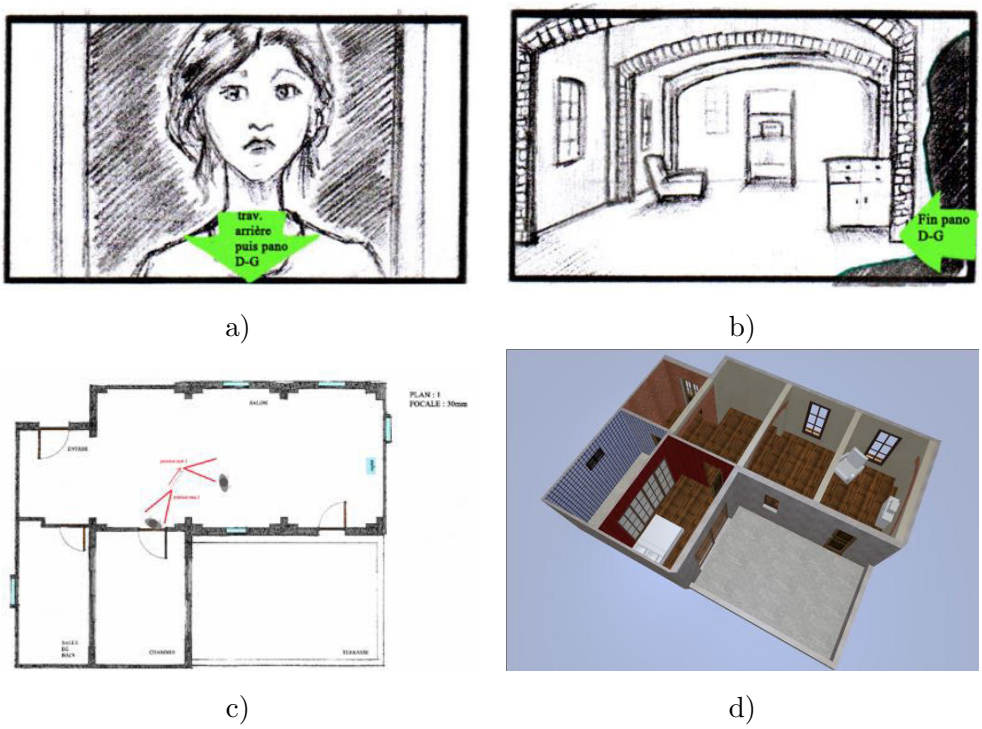


Fig. A.1: *a) First frame of the shot. b) Last frame of the shot. c) Floor-plan view with director's annotations showing camera and actors displacements. Traditional storyboards are useful for placing actors and cameras, but provide little support for stereoscopic 3-D. We use them as input for previz. d) Blender 3D model of the "Endless Night" apartment.*

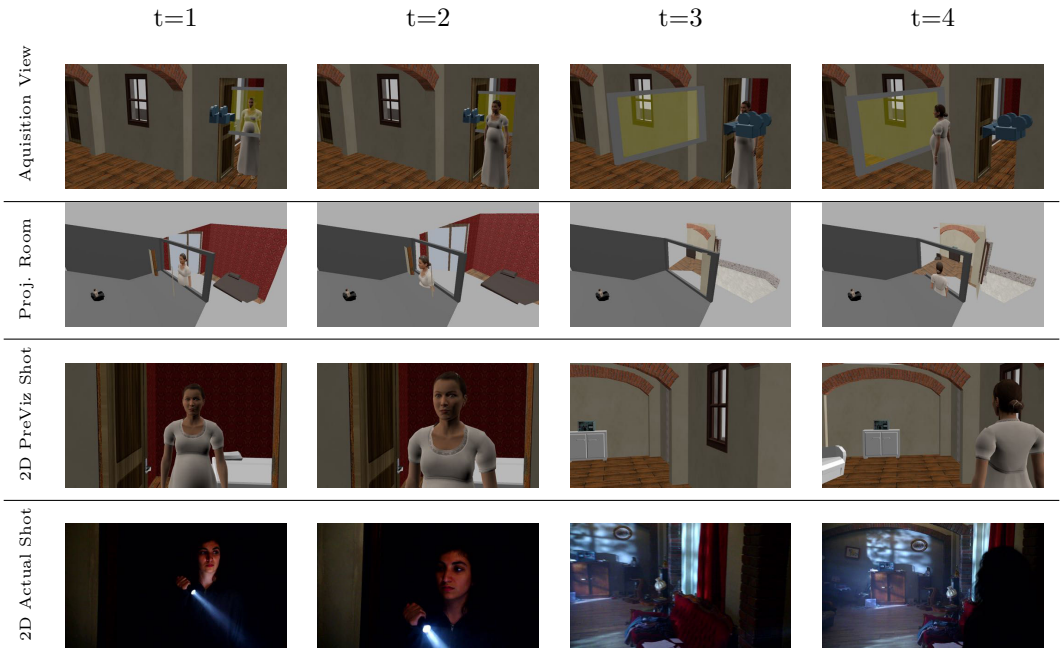


Fig. A.2: *Previz results for a panning shot. Columns show different times in the shot, arranged chronologically from left to right. First row: acquisition view; second row: virtual projection room; third row: previz results; bottom row: actual rushes. The actress enters the hall and the camera pans to the left. The difficulty of this shot is to handle the transition from a narrow single subject (the actress) with a small depth volume into a room with a bigger depth volume. The stereoscopic parameters are dynamically adjusted across the shot to create the desired stereoscopic effect.*

B Super-Resolved Generated Images

B.1 Results

We present the full resolution images corresponding to the closeups in Fig. 4.9 in Chapter 4 reproduced here as Fig. B.1. In Fig. B.2 to B.7, we show for each data set the ground-truth image (if available), the disparity map used for novel view synthesis, the view generated with the approach in Wanner and Goldluecke (2012), as well as the view generated by the proposed method. Results of the previous method where obtained using the public released implementation of the code available at
http://sourceforge.net/projects/cocolib/.

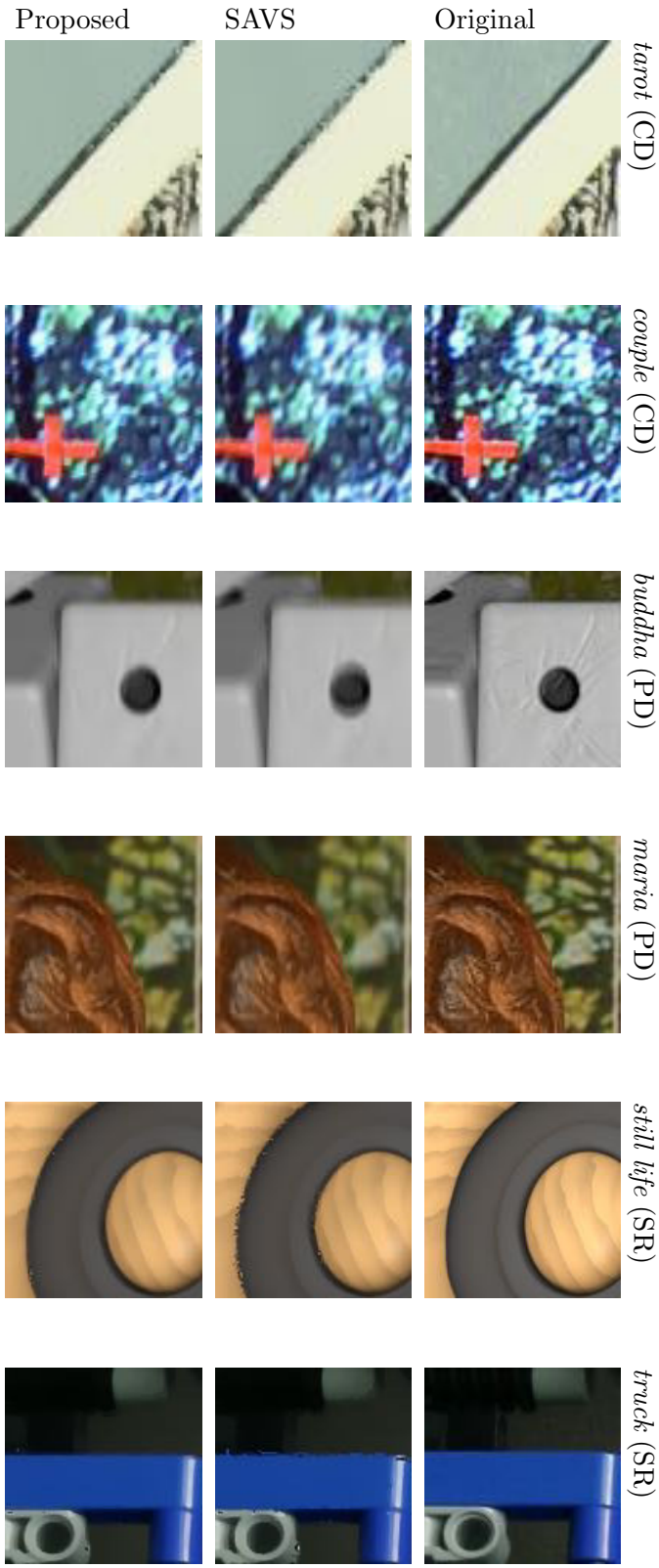
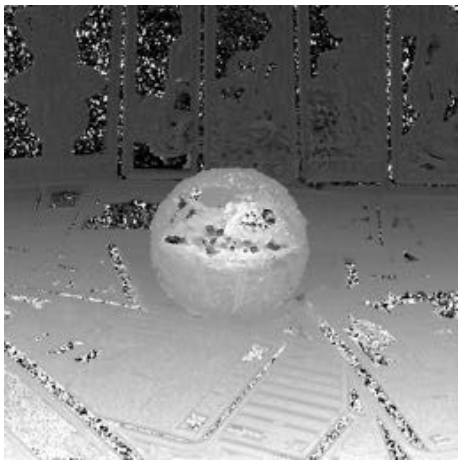


Fig. B.1: *Visual comparison of novel views obtained for different light fields. From top to bottom, the rows present closeups of the ground truth images (Original), the results obtained by Wanner and Goldluecke (2012) (SAVS), and our results (Proposed). CD stands for computed disparity, PD for planar disparity and SR for super-resolution, see text for details. The results obtained by the proposed method are visibly sharper, in particular along color edges.*

Original

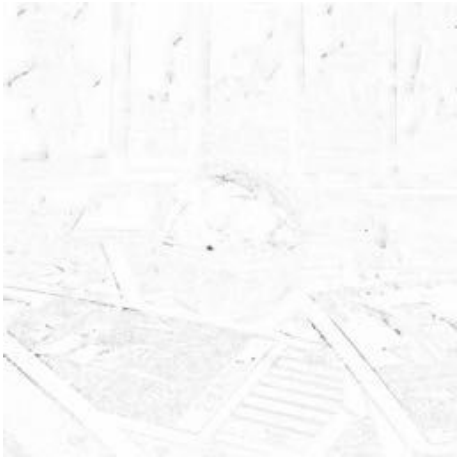
Estimated disparity map



Previous

Proposed

Previous Diff



Proposed Diff

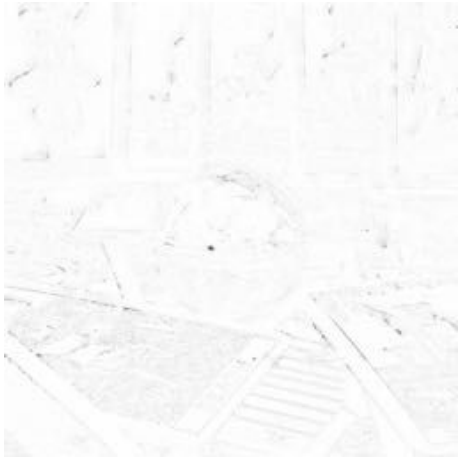
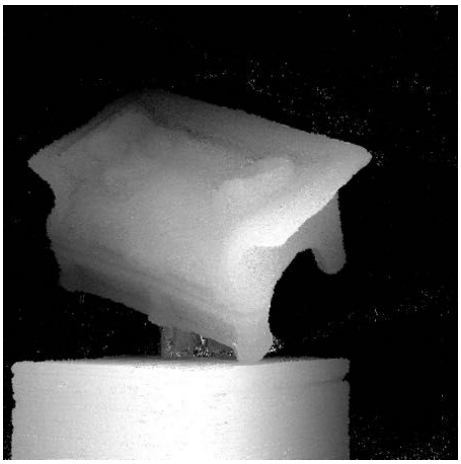


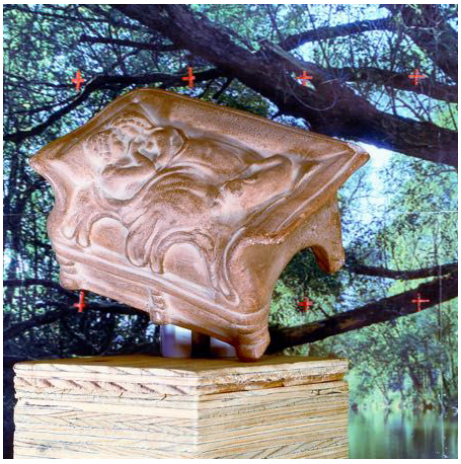
Fig. B.2: *Novel view of the Stanford gantry data set "Tarot" (fine configuration). Synthesized at x1 resolution using the estimated disparity map.*

Original

Estimated disparity map



Previous



Proposed

Previous Diff

Proposed Diff

Fig. B.3: *Novel view of the HCI gantry data set "Couple". Synthesized at x1 resolution using the estimated disparity map.*

Original

Flat geometric proxy

Previous

Proposed

Previous Diff

Proposed Diff

Fig. B.4: *Novel view of the HCI raytraced data set "Buddha". Synthesized at x1 resolution using a plane in the center of the scene as geometric proxy.*

Original

Flat geometric proxy

Previous

Proposed

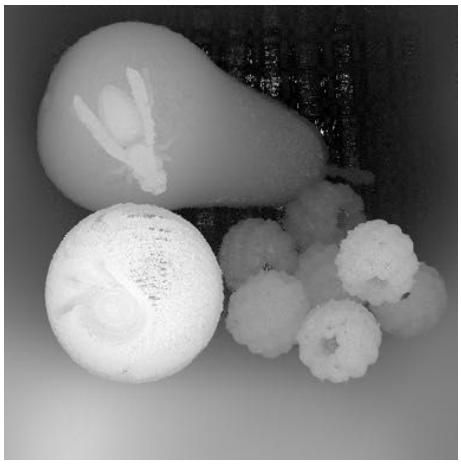
Previous Diff

Proposed Diff

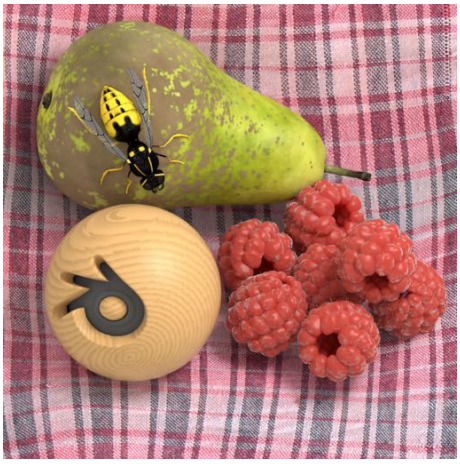
Fig. B.5: *Novel view of the HCI gantry data set "Maria". Synthesized at x1 resolution using a plane in the center of the scene as geometric proxy.*

Original

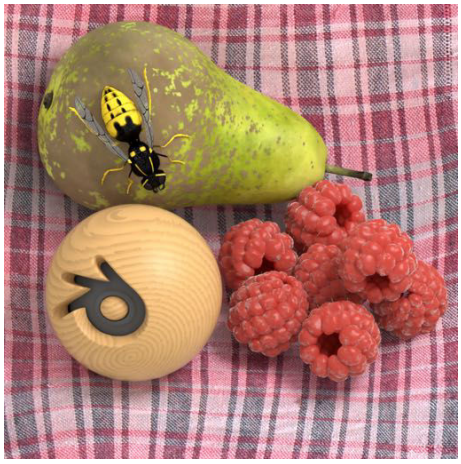
Estimated disparity map



Previous



Proposed



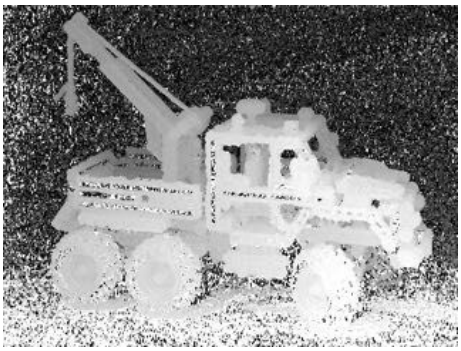
Previous Diff

Proposed Diff

Fig. B.6: *Novel view of the HCI gantry data set "Still life". Synthesized at x3 resolution using the estimated disparity map.*

Original

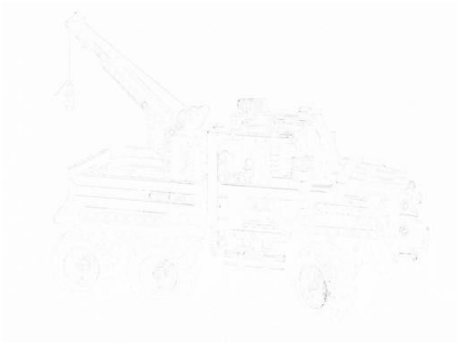
Estimated disparity map



Previous

Proposed

Previous Diff



Proposed Diff

Fig. B.7: *Novel view of the Stanford gantry data set "Truck". Synthesized at x3 resolution using the estimated disparity map.*

C

Results from Unstructured Camera Configurations

We present the full resolution images corresponding to the closeups in Fig. 4.15 in chapter 4 reproduced here as Fig. C.1. The groundtruth target images is labeled as "Original", the results obtained by Wanner and Goldluecke (2012) are labeled "SAVS", the results obtained by Buehler *et al.* (2001) are labeled "ULR", and our results are labeled "Proposed". G1, G3 stand for different geometric reconstructions described in Sec. 4.5.3

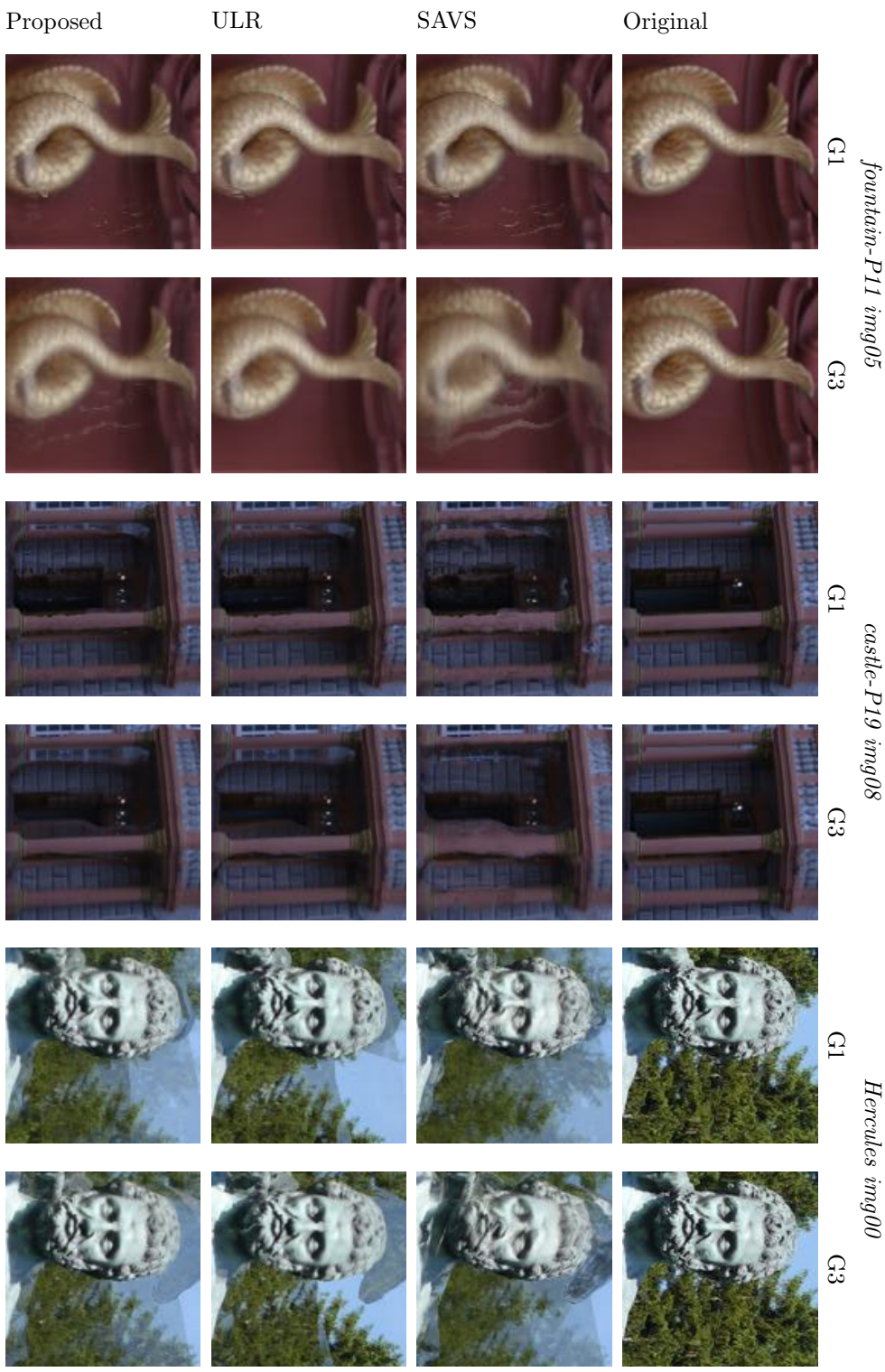


Fig. C.1: *Visual comparison of the novel views obtained for different datasets. From top to bottom, the rows present closeups of the Original view, the results obtained by Wanner and Goldluecke (2012) (SAVS), the results obtained by Buehler et al. (2001) (ULR), and our results (Proposed). G1, G3 stand for different geometric reconstructions described in Sec. 4.5.3.*

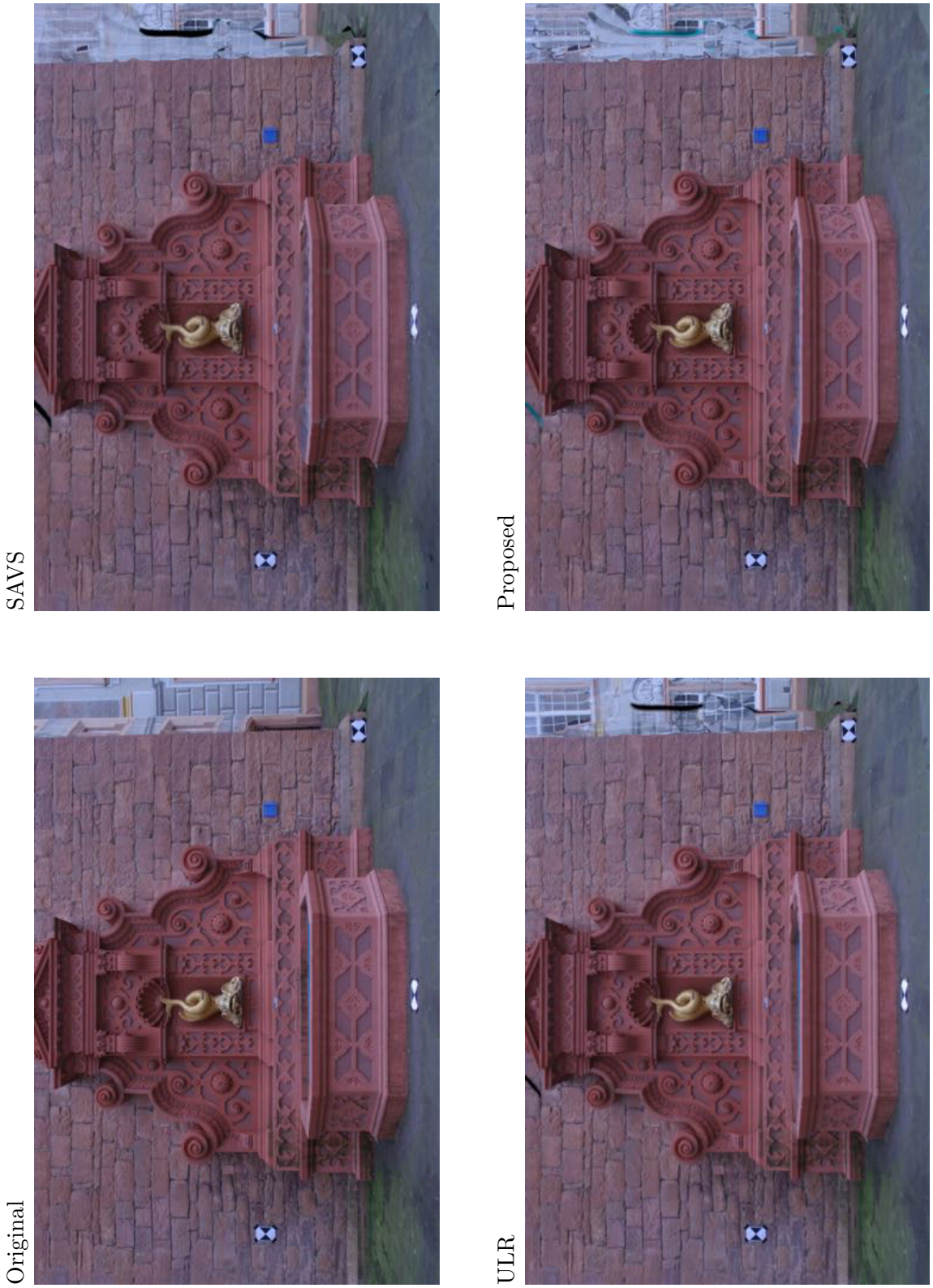


Fig. C.2: *Full resolution images obtained for the image 05 of the "fountain-P11" dataset using the geometry G1.*

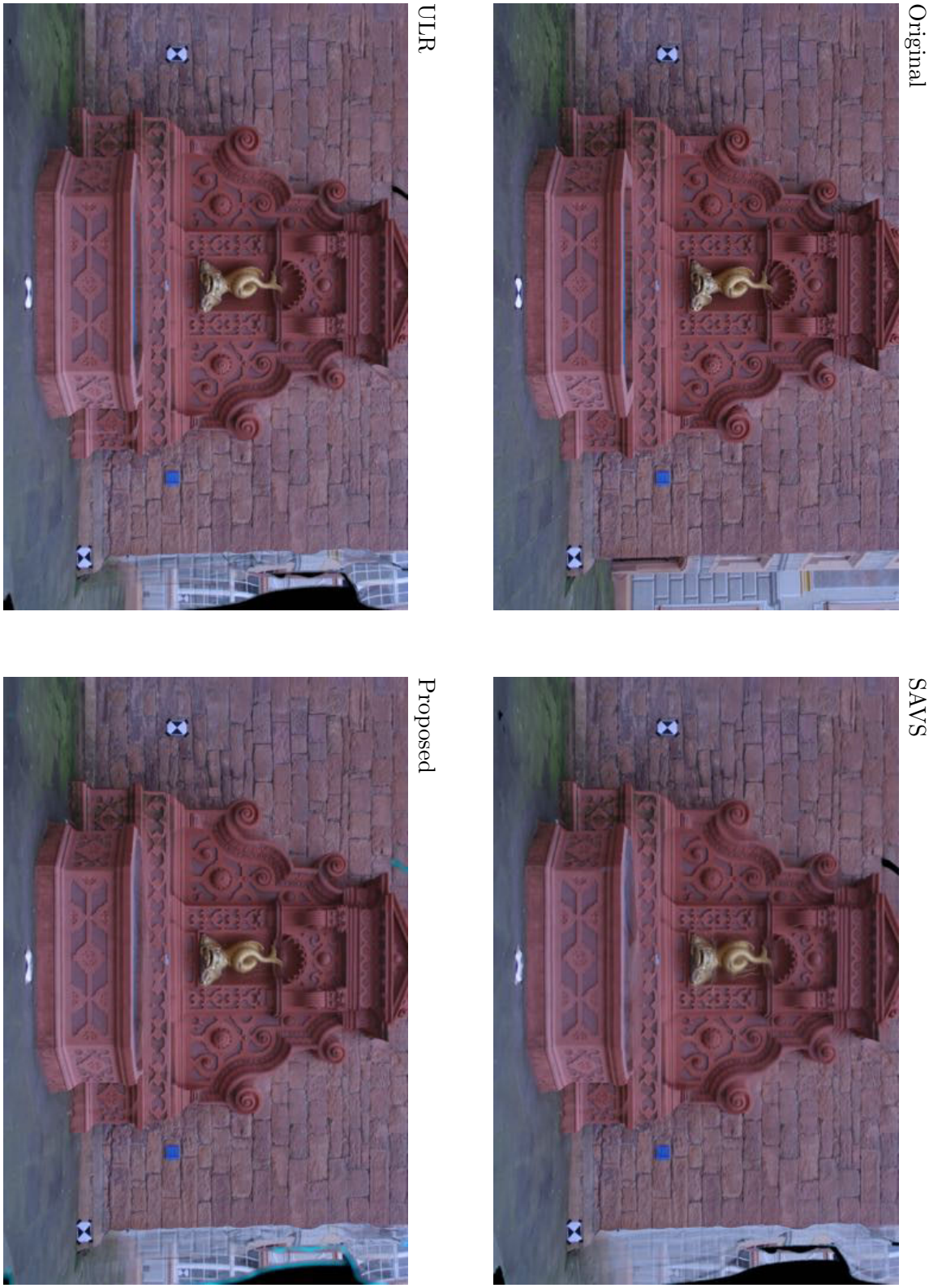


Fig. C.3: *Full resolution images obtained for the image 05 of the "fountain-P11" dataset using the geometry G3.*

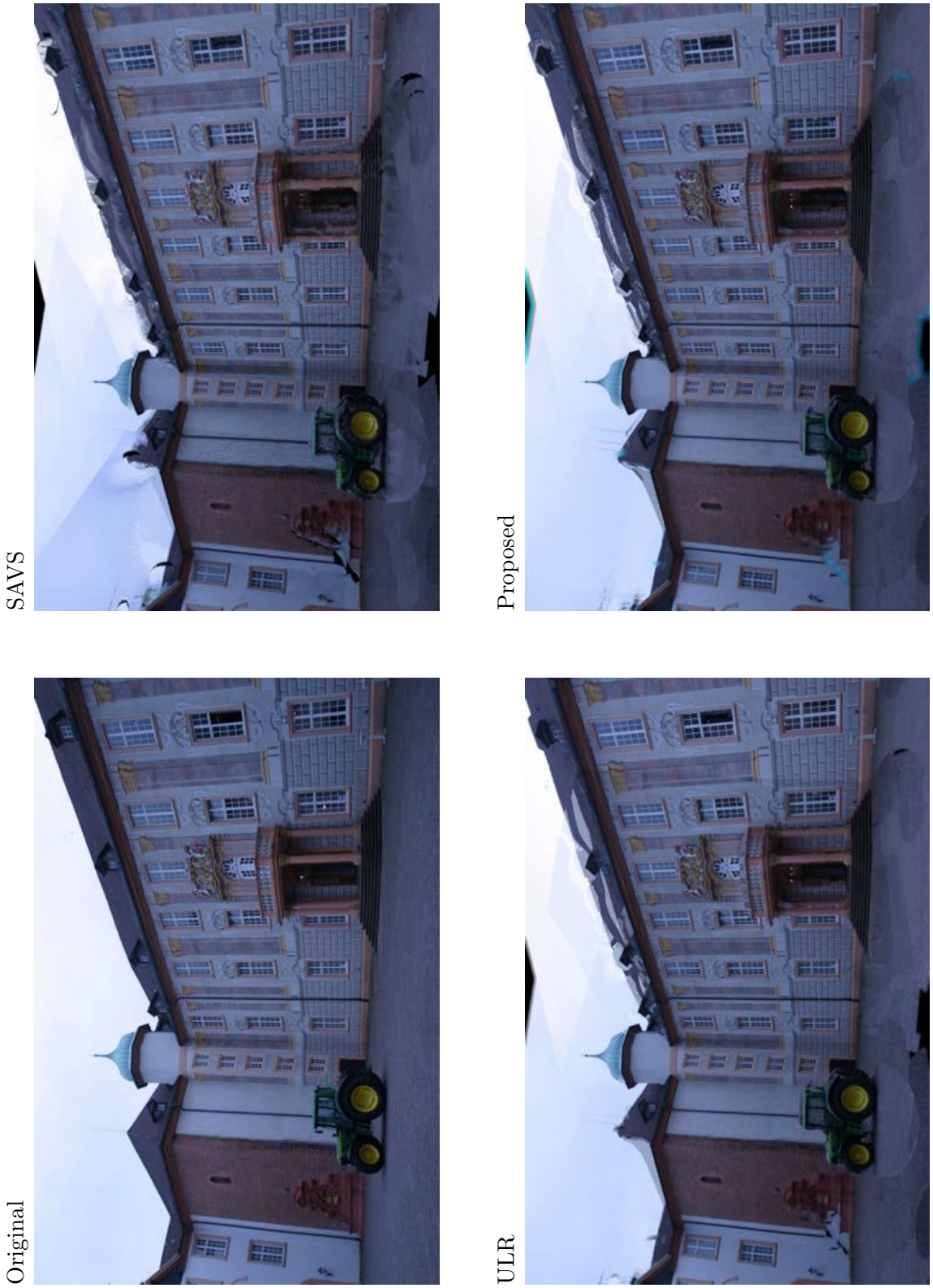


Fig. C.4: *Full resolution images obtained for the image 08 of the "castle-P19" dataset using the geometry G1.*

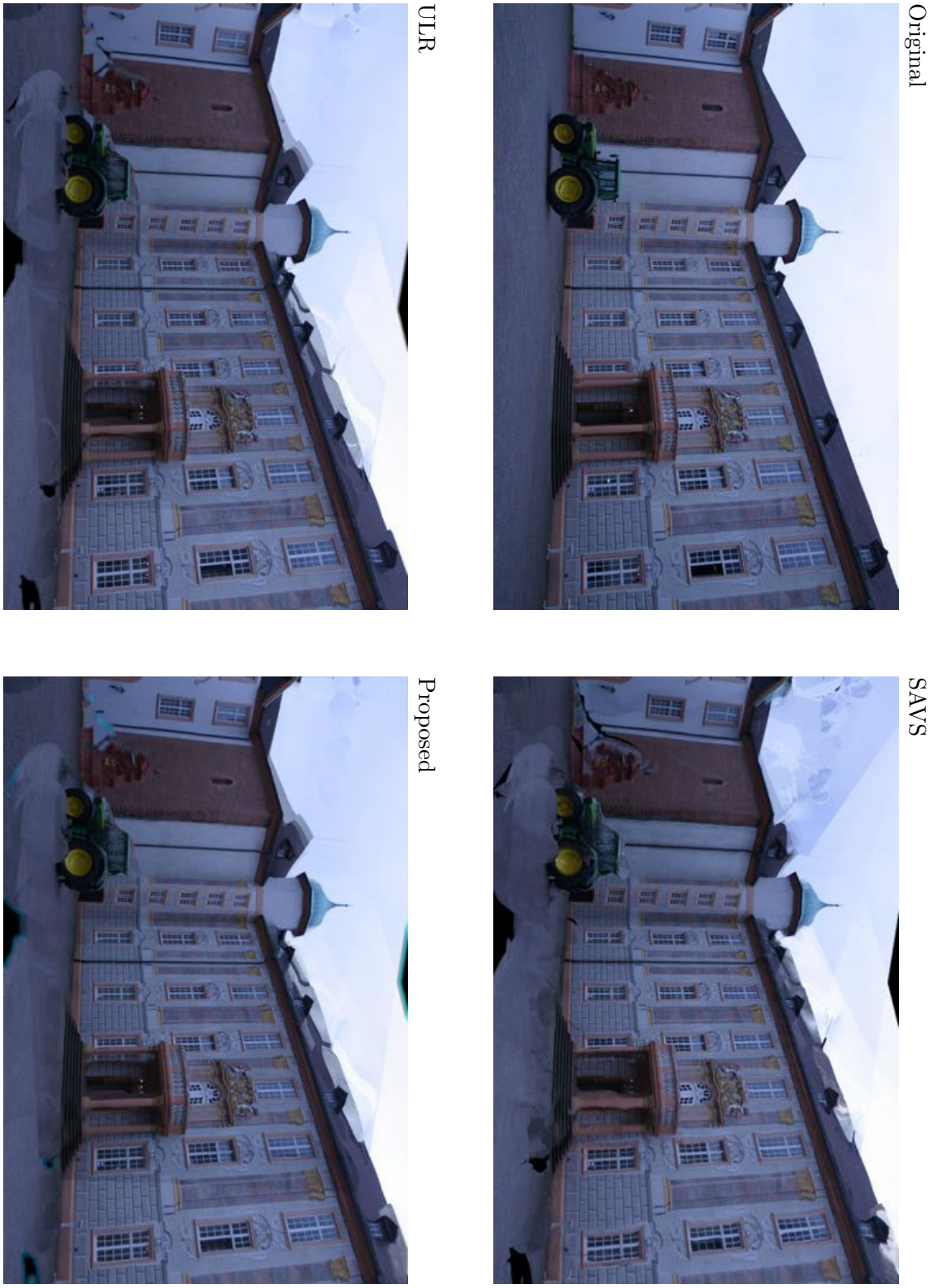


Fig. C.5: *Full resolution images obtained for the image 08 of the "castle-P19" dataset using the geometry G3.*

Original

SAVS

ULR

Proposed

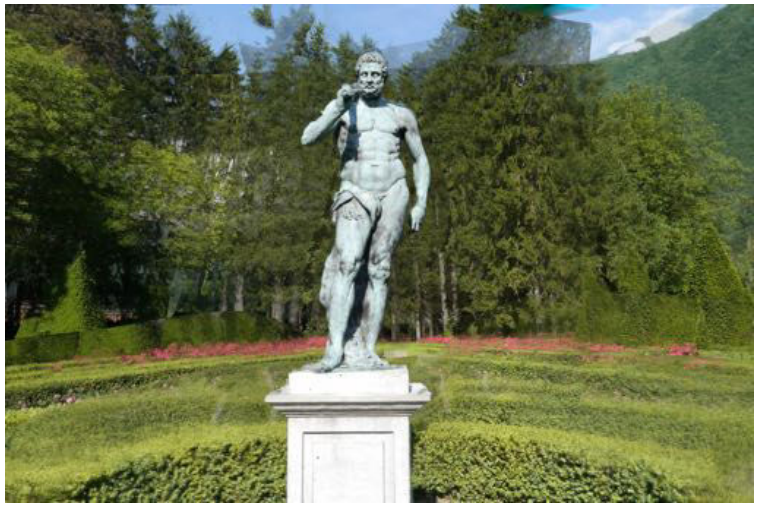


Fig. C.6: *Full resolution images obtained for the image 00 of the "Hercules" dataset using the geometry G1.*

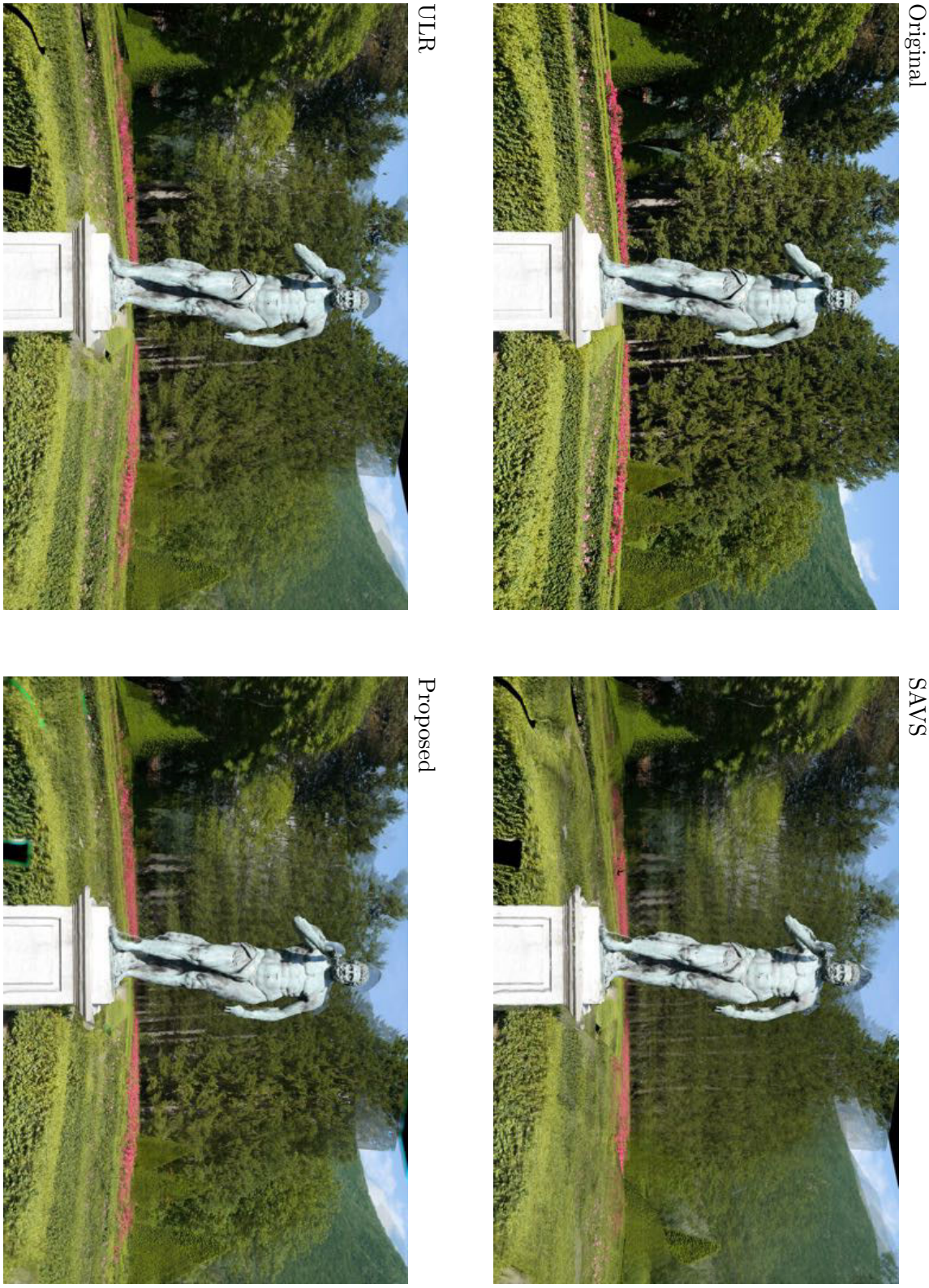


Fig. C.7: *Full resolution images obtained for the image 00 of the "Hercules" dataset using the geometry G3.*

Bibliography

3ality Technica. "3ality Technica". http://www.3alitytechnica.com/3D-rigs/index.php (2015). [Online; accessed 14-August-2015]. 151

Adelson, E.H. and Bergen, J.R. "The plenoptic function and the elements of early vision". In M.S. Landy and J.A. Movshon, editors, "Computational Models of Visual Processing", pages 3–20. MIT Press (1991). 58

Agarwala, A., Dontcheva, M., Agrawala, M., Drucker, S., Colburn, A., Curless, B., Salesin, D. and Cohen, M. "Interactive digital photomontage". In "SIGGRAPH", pages 294–302. ACM (2004). doi:10.1145/1186562.1015718. 62, 108

Akenine-Möller, T., Haines, E. and Hoffman, N. *Real-Time Rendering 3rd Edition*. A. K. Peters, Ltd. (2008). ISBN 987-1-56881-424-7. 75

Akhter, R., Sazzad, Z.P., Horita, Y. and Baltes, J. "No-reference stereoscopic image quality assessment". In "Stereoscopic Displays and Applications XXI", pages 75240T–75240T–12. SPIE (2010). doi:10.1117/12.838775. 149, 157

Allison, R. "Analysis of the influence of vertical disparities arising in toed-in stereoscopic cameras". In *Journal of Imaging Science and Technology*, 51(4):317–327 (2007). doi:10.2352/J.ImagingSci.Technol.(2007)51:4(317). 12

Angenieux. "Online angenieux portefolio". http://www.angenieux.com/zoom-lenses/cinema-portfolio/optimo-28-340.htm (2015). [Online; accessed 14-August-2015]. 49

Baker, S. and Kanade, T. "Limits on super-resolution and how to break them". In *Transactions on Pattern Analysis and Machine Intelligence*, 24(9):1167–1183 (2002). doi:10.1109/TPAMI.2002.1033210. 73

Baker, S. and Matthews, I. "Lucas-kanade 20 years on: A unifying framework". In *International Journal of Computer Vision*, 56(3):221–255 (2004). doi:10.1023/B:VISI.0000011205.11775.fd. 42

Banks, M.S., Cooper, E.A. and Piazza, E.A. "Camera focal length and the perception of pictures". In *Ecological Psychology*, 26(1-2):30–46 (2014). doi:10.1080/10407413.2014.877284. 37, 53

Banks, M.S., Kim, J. and Shibata, T. "Insight into vergence-accommodation mismatch". In "Head- and Helmet-Mounted Displays XVIII: Design and Applications", pages 873509–873509–12. SPIE (2013). doi:10.1117/12.2019866. 12, 14, 32, 47, 49

Bay, H., Ess, A., Tuytelaars, T. and Van Gool, L. "Speeded-up robust features (SURF)". In *Computer Vision and Image Understanding*, 110(3):346–359 (2008). doi:10.1016/j.cviu.2007.09.014. 86

Beck, A. and Teboulle, M. "A fast iterative shrinkage-thresholding algorithm for linear inverse problems". In *Journal on Imaging Sciences*, 2(1):183–202 (2009). doi:10.1137/080716542. 74

Beier, T. and Neely, S. "Feature-based image metamorphosis". In "SIGGRAPH", pages 35–42. ACM (1992). doi:10.1145/133994.134003. 58

Binocle. "Binocle web page". http://binocle.com (2015). [Online; accessed 14-August-2015]. 151

Bishop, T. and Favaro, P. "The light field camera: Extended depth of field, aliasing, and superresolution". In *Transactions on Pattern Analysis and Machine Intelligence*, 34(5):972–986 (2012). doi:10.1109/TPAMI.2011.168. 63

Blender. "Blender". http://www.blender.org/ (2015). [Online; accessed 14-August-2015]. 37, 159

Bonfort, T. and Sturm, P. "Voxel carving for specular surfaces". In "International Conference on Computer Vision", pages 591–596. IEEE (2003). doi:10.1109/ICCV.2003.1238401. 64

Bordwell, D. *On the history of film style*. Harvard University Press (1997). ISBN 978-0674634299. 1

Buehler, C., Bosse, M., McMillan, L., Gortler, S. and Cohen, M. "Unstructured lumigraph rendering". In "SIGGRAPH", pages 425–432. ACM (2001). doi:10.1145/383259.383309. 4, 5, 55, 56, 57, 59, 61, 62, 63, 72, 74, 79, 91, 92, 100, 103, 104, 105, 107, 108, 109, 110, 111, 112, 113, 130, 131, 133, 135, 142, 144, 146, 148, 149, 154, 156, 171, 172

Canudo, R. "The birth of a sixth art". In R. Abel, editor, "French Film Theory and Criticism: 1907-1929", pages 58–66. Princeton University Press (1993). 1

Carranza, J., Theobalt, C., Magnor, M.A. and Seidel, H.P. "Free-viewpoint video of human actors". In "SIGGRAPH", pages 569–577. ACM (2003). doi: 10.1145/1201775.882309. 60

Chambolle, A. "An algorithm for total variation minimization and applications". In *Journal of Mathematical Imaging and Vision*, 20(1-2):89–97 (2004). doi:10.1023/B:JMIV.0000011325.36760.1e. 73

Chang, C.H., Liang, C.K. and Chuang, Y.Y. "Content-aware display adaptation and interactive editing for stereoscopic images". In *Transactions on Multimedia*, 13(4):589–601 (2011). doi:10.1109/TMM.2011.2116775. 44, 49

Chang, C.F., Bishop, G. and Lastra, A. "LDI tree: A hierarchical representation for image-based rendering". In "SIGGRAPH", pages 291–298. ACM (1999). doi:10.1145/311535.311571. 58

Chaurasia, G., Duchene, S., Sorkine-Hornung, O. and Drettakis, G. "Depth synthesis and local warps for plausible image-based navigation". In *Transactions on Graphics*, 32(3):30 (2013). doi:10.1145/2487228.2487238. 62

Chauvier, L., Murray, K., Parnall, S., Taylor, R. and Walker, J. "Does size matter? the impact of screen size on stereoscopic 3DTV". In "IBC conference", (2010). 35

Chen, M.J., Cormack, L.K. and Bovik, A.C. "No-reference quality assessment of natural stereopairs". In *Transactions on Image Processing*, 22(9):3379–3391 (2013). doi:10.1109/TIP.2013.2267393. 149, 157

Chen, S.E. and Williams, L. "View interpolation for image synthesis". In "SIGGRAPH", pages 279–288. ACM (1993). doi:10.1145/166117.166153. 58, 59, 84

Chen, W. *Multidimensional characterization of quality of experience of stereoscopic 3D TV*. Theses, Université de Nantes Angers Le Mans (2012). 49

Cheng, Y. "Mean shift, mode seeking, and clustering". In *Transactions on Pattern Analysis and Machine Intelligence*, 17(8):790–799 (1995). doi:10.1109/34.400568. 108

Cho, T.S., Zitnick, C.L., Joshi, N., Kang, S.B., Szeliski, R. and Freeman, W.T. "Image restoration by matching gradient distributions". In *Transactions on Pattern Analysis and Machine Intelligence*, 34(4):683–694 (2012). doi:10.1109/TPAMI.2011.166. 73

Cumming, B. and Judge, S. "Disparity-induced and blur-induced convergence eye movement and accommodation in the monkey". In *Journal of neurophysiology*, 55(5):896–914 (1986). 14

Davis, A., Levoy, M. and Durand, F. "Unstructured light fields". In *Computer Graphics Forum*, 31(2pt1):305–314 (2012). doi:10.1111/j.1467-8659.2012.03009.x. 58, 103

Debevec, P., Yu, Y. and Borshukov, G. "Efficient view-dependent image-based rendering with projective texture-mapping". In "Rendering Techniques (Eurographics)", pages 105–116. Springer (1998). doi:10.1007/978-3-7091-6453-2_10. 58, 62

Debevec, P.E., Taylor, C.J. and Malik, J. "Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach". In "SIGGRAPH", pages 11–20. ACM (1996). doi:10.1145/237170.237191. 62

Devernay, F. *Stereoscopic vision and differential properties of surfaces.* Theses, Ecole Polytechnique X (1997). 25

Devernay, F. and Beardsley, P. "Stereoscopic cinema". In R. Ronfard and G. Taubin, editors, "Image and Geometry Processing for 3D Cinematography", pages 11–51. Springer Berlin Heidelberg (2010). 7, 8, 9, 11, 12, 26, 27, 28, 32, 35, 52

Devernay, F. and Duchêne, S. "New view synthesis for stereo cinema by hybrid disparity remapping". In "International Conference on Image Processing", pages 5–8. IEEE (2010). doi:10.1109/ICIP.2010.5649194. 35, 42, 48, 54, 147

Devernay, F., Duchêne, S. and Ramos-Peon, A. "Adapting stereoscopic movies to the viewing conditions using depth-preserving and artifact-free novel view synthesis". In "Stereoscopic Displays and Applications XXII", pages 786302–786302. SPIE (2011). doi:10.1117/12.872883. 43

Didyk, P., Ritschel, T., Eisemann, E., Myszkowski, K. and Seidel, H.P. "Adaptive image-space stereo view synthesis". In "Vision, Modeling and Visualization Workshop", pages 299–306 (2010). 50

Diebel, J. "Representing attitude: Euler angles, unit quaternions, and rotation vectors". In *Matrix*, 58:15–16 (2006). 18

Dsouza, C. *Think in 3D: Food For Thought for Directors, Cinematographers and Stereographers.* CreateSpace Independent Publishing Platform (2012). ISBN 978-1470150778. 52, 54

Eisemann, M., De Decker, B., Magnor, M., Bekaert, P., de Aguiar, E., Ahmed, N., Theobalt, C. and Sellent, A. "Floating textures". In *Computer Graphics Forum*, 27(2):409–418 (2008). doi:10.1111/j.1467-8659.2008.01138.x. 61

Faugeras, O. *Three-dimensional computer vision: a geometric viewpoint.* MIT press (1993). ISBN 9780262061582. xi, 18

Faugeras, O. and Luong, Q.T. *The geometry of multiple images: the laws that govern the formation of multiple images of a scene and some of their applications.* MIT press (2004). ISBN 9780262562041. xi, 21

Fincham, E. and Walton, J. "The reciprocal actions of accommodation and convergence". In *The Journal of Physiology*, 137(3):488–508 (1957). doi:10.1113/jphysiol.1957.sp005829. 14

Fischler, M.A. and Bolles, R.C. "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography". In *Communications of the ACM*, 24(6):381–395 (1981). doi:10.1145/358669.358692. 42, 108

Fitzgibbon, A., Wexler, Y. and Zisserman, A. "Image-based rendering using image-based priors". In *International Journal of Computer Vision*, 63(2):141–151 (2005). doi:10.1007/s11263-005-6643-9. 60, 108

Forsyth, D.A. and Ponce, J. *Computer Vision: A Modern Approach*. Prentice Hall Professional Technical Reference (2002). xi, 18, 20

Furukawa, Y., Curless, B., Seitz, S.M. and Szeliski, R. "Reconstructing building interiors from images". In "International Conference on Computer Vision", pages 80–87. IEEE (2009). doi:10.1109/ICCV.2009.5459145. 65

Furukawa, Y., Curless, B., Seitz, S.M. and Szeliski, R. "Towards internet-scale multi-view stereo". In "Conference on Computer Vision and Pattern Recognition", pages 1434–1441. IEEE (2010). doi:10.1109/CVPR.2010.5539802. 64, 85, 93, 100

Furukawa, Y. and Ponce, J. "Carved visual hulls for image-based modeling". In "European Conference on Computer Vision", pages 564–577. Springer (2006). doi:10.1007/11744023_44. 64

Furukawa, Y. and Ponce, J. "Accurate, dense, and robust multiview stereopsis". In *Transactions on Pattern Analysis and Machine Intelligence*, 32(8):1362–1376 (2010). doi:10.1109/TPAMI.2009.161. 64, 85, 90, 93, 157

Fusiello, A., Roberto, V. and Trucco, E. "Efficient stereo with multiple windowing". In "Conference on Computer Vision and Pattern Recognition", pages 858–863. IEEE (1997). doi:10.1109/CVPR.1997.609428. 65

Fusiello, A., Trucco, E. and Verri, A. "A compact algorithm for rectification of stereo pairs". In *Machine Vision and Applications*, 12(1):16–22 (2000). doi:10.1007/s001380050120. 21

Gallup, D., Frahm, J.M. and Pollefeys, M. "Piecewise planar and non-planar stereo for urban scene reconstruction". In "Conference on Computer Vision and Pattern Recognition", pages 1418–1425. IEEE (2010). doi:10.1109/CVPR.2010.5539804. 85

Gargallo, P., Sturm, P. and Pujades, S. "An occupancy-depth generative model of multi-view images". In "Asian Conference on Computer Vision", pages 373–383. Springer (2007). doi:10.1007/978-3-540-76390-1_37. 65

Gargallo I Piracés, P. *Contributions to the Bayesian Approach to Multi-View Stereo*. Theses, Institut National Polytechnique de Grenoble - INPG (2008). 67

Germann, M., Hornung, A., Keiser, R., Ziegler, R., Würmlin, S. and Gross, M. "Articulated billboards for video-based rendering". In *Computer Graphics Forum*, 29(2):585–594 (2010). doi:10.1111/j.1467-8659.2009.01628.x. 61

Germann, M., Popa, T., Keiser, R., Ziegler, R. and Gross, M. "Novel-view synthesis of outdoor sport events using an adaptive view-dependent geometry". In *Computer Graphics Forum*, 31(2pt1):325–333 (2012). doi:10.1111/j.1467-8659.2012.03011.x. 61

Gibson, J.J. *The perception of the visual world*. Houghton Mifflin (1950). 7, 8

Girod, B., Greiner, G. and Niemann, H. *Principles of 3D image analysis and synthesis*. Springer (2000). xi

Goesele, M., Ackermann, J., Fuhrmann, S., Haubold, C., Klowsky, R., Steedly, D. and Szeliski, R. "Ambient point clouds for view interpolation". In "SIGGRAPH", pages 95:1–95:6. ACM (2010). doi:10.1145/1833349.1778832. 60

Goldluecke, B. and Cremers, D. "Superresolution texture maps for multiview reconstruction". In "International Conference on Computer Vision", pages 1677 – 1684. IEEE (2009). doi:10.1109/ICCV.2009.5459378. 63

Gortler, S.J., Grzeszczuk, R., Szeliski, R. and Cohen, M.F. "The Lumigraph". In "SIGGRAPH", pages 43–54. ACM (1996). doi:10.1145/237170.237200. 4, 5, 58, 62, 79, 131, 154

Grau, O., Thomas, G.A., Hilton, A., Kilner, J. and Starck, J. "A robust free-viewpoint video system for sport scenes". In "3DTV Conference", pages 1–4. IEEE (2007). doi:10.1109/3DTV.2007.4379384. 61

Guillemaut, J.Y. and Hilton, A. "Joint multi-layer segmentation and reconstruction for free-viewpoint video applications". In *International Journal of Computer Vision*, 93(1):73–100 (2011). doi:10.1007/s11263-010-0413-z. 61

Guillemaut, J.Y., Kilner, J. and Hilton, A. "Robust graph-cut scene segmentation and reconstruction for free-viewpoint video of complex dynamic scenes". In "International Conference on Computer Vision", pages 809–816. IEEE (2009). doi:10.1109/ICCV.2009.5459299. 61

Guo, C., Ma, Q. and Zhang, L. "Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform". In "Conference on Computer Vision and Pattern Recognition", pages 1–8. IEEE (2008). doi:10.1109/CVPR.2008.4587715. 43

Guo, Y., Liu, F., Shi, J., Zhou, Z.H. and Glei, M. "Image retargeting using mesh parametrization". In *Transactions on Multimedia*, 11(5):856–867 (2009). 43

Hartley, R.I. and Zisserman, A. *Multiple View Geometry in Computer Vision.* Cambridge University Press, second edition (2004). ISBN 0521540518. xi, 18, 19, 21, 85, 86

Hartley, R. and Kang, S.B. "Parameter-free radial distortion correction with center of distortion estimation". In *Transactions on Pattern Analysis and Machine Intelligence*, 29(8):1309–1321 (2007). doi:10.1109/TPAMI.2007.1147. 99

Hayashi, K. and Saito, H. "Synthesizing free-viewpoing images from multiple view videos in soccer stadium". In "Computer Graphics, Imaging and Visualisation", pages 220–225. IEEE (2006). doi:10.1109/CGIV.2006.83. 61

Heckbert, P.S. *Fundamentals of texture mapping and image warping.* theses, University of California, Berkeley (1989). 74, 75, 76

Hegel, G.W.F. *Lectures on aesthetics.* Heinrich Gustav Hotho, Berlin (1835). 1

Heigl, B., Koch, R., Pollefeys, M., Denzler, J. and Van Gool, L. "Plenoptic modeling and rendering from image sequences taken by a hand-held camera". In "Mustererkennung", pages 94–101. Springer (1999). doi: 10.1007/978-3-642-60243-6_11. 62

Held, R.T. and Banks, M.S. "Misperceptions in stereoscopic displays: A vision science perspective". In "Proceedings of the 5th symposium on Applied perception in graphics and visualization", pages 23–32. ACM (2008). doi: 10.1145/1394281.1394285. 26, 155

Held, R.T., Cooper, E.A., O'brien, J.F. and Banks, M.S. "Using blur to affect perceived distance and size". In *Transactions on Graphics*, 29(2):19:1–19:16 (2010). doi:10.1145/1731047.1731057. 8

Heuel, S. *Uncertain projective geometry: statistical reasoning for polyhedral object reconstruction.* Springer (2004). doi:10.1007/b97201. 86

Hilton, A., Guillemaut, J.Y., Kilner, J., Grau, O. and Thomas, G. "3D-TV production from conventional cameras for sports broadcast". In *Transactions on Broadcasting*, 57(2):462–476 (2011). doi:0.1109/TBC.2011.2131870. 61, 134

Hirschmüller, H. "Stereo processing by semiglobal matching and mutual information". In *Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341 (2008). doi:10.1109/TPAMI.2007.1166. 42

Hoffman, D.M., Girshick, A.R., Akeley, K. and Banks, M.S. "Vergence–accommodation conflicts hinder visual performance and cause visual fatigue". In *Journal of Vision*, 8(3):33 (2008). doi:10.1167/8.3.33. 12, 14

Hofsetz, C., Ng, K., Chen, G., McGuinness, P., Max, N. and Liu, Y. "Image-based rendering of range data with estimated depth uncertainty". In *Computer Graphics and Applications*, 24(4):34–41 (2004). doi:10.1109/MCG.2004.8. 59, 65

Hornung, A. and Kobbelt, L. "Interactive pixel-accurate free viewpoint rendering from images with silhouette aware sampling". In *Computer Graphics Forum*, 28(8):2090–2103 (2009). doi:10.1111/j.1467-8659.2009.01416.x. 61, 62

Howard, I. and Rogers, B. *Seeing in Depth*. Oxford University Press (2008). doi:10.1093/acprof:oso/9780195367607.001.0001. 125

Hu, X. and Mordohai, P. "A quantitative evaluation of confidence measures for stereo vision". In *Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2121–2133 (2012). doi:10.1109/TPAMI.2012.46. 65

Jantet, V., Guillemot, C. and Morin, L. "Joint projection filling method for occlusion handling in depth-image-based rendering". In *3D Research*, 2(4):1–13 (2011). doi:10.1007/3DRes.04(2011)4. 43

Kanade, T., Rander, P. and Narayanan, P. "Virtualized reality: constructing virtual worlds from real scenes". In *MultiMedia*, 4(1):34–47 (1997). doi:10.1109/93.580394. 60

Kanade, T. and Okutomi, M. "A stereo matching algorithm with an adaptive window: Theory and experiment". In *Transactions on Pattern Analysis and Machine Intelligence*, 16(9):920–932 (1994). doi:10.1109/34.310690. 65

Kazhdan, M., Bolitho, M. and Hoppe, H. "Poisson surface reconstruction". In "Eurographics symposium on Geometry processing", volume 7, pages 61–70. Eurographics Association (2006). 65, 85, 90, 93, 157

Kim, C., Hornung, A., Heinzle, S., Matusik, W. and Gross, M. "Multi-perspective stereoscopy from light fields". In "SIGGRAPH Asia", pages 190:1–190:10. ACM (2011). doi:10.1145/2024156.2024224. 54

Kim, H. and Hilton, A. "Environment modelling using spherical stereo imaging". In "International Conference on Computer Vision Workshops", pages 1534–1541. IEEE (2009). doi:10.1109/ICCVW.2009.5457429. 65

Kim, H.J., Choi, J.W., Chang, A.J. and Yu, K.Y. "Reconstruction of stereoscopic imagery for visual comfort". In "Stereoscopic Displays and Applications XIX", pages 680303–680303. SPIE (2008). doi:10.1117/12.766395. 42

Kooi, F.L. and Toet, A. "Visual comfort of binocular and 3D displays". In *Displays*, 25(2-3):99–108 (2004). doi:10.1016/j.displa.2004.07.004. 12

Kopf, J., Cohen, M.F. and Szeliski, R. "First-person hyper-lapse videos". In "SIGGRAPH", pages 78:1–78:10. ACM (2014). doi:10.1145/2601097.2601195. 62, 79, 98, 103

Kowalczuk, J., Psota, E.T. and Perez, L.C. "Real-time stereo matching on CUDA using an iterative refinement method for adaptive support-weight correspondences". In *Transactions on Circuits and Systems for Video Technology*, 23(1):94–104 (2013). doi:10.1109/TCSVT.2012.2203200. 44

Kutulakos, K.N. and Seitz, S.M. "A theory of shape by space carving". In *International Journal of Computer Vision*, 38(3):199–218 (2000). doi:10.1023/A:1008191222954. 64

Lambooij, M.T., IJsselsteijn, W.A., Fortuin, M. and Heynderickx, I. "Visual discomfort and visual fatigue of stereoscopic displays: A review". In , 53(3):1–14 (2009). doi:10.2352/J.ImagingSci.Technol.2009.53.3.030201. 14, 15

Lambooij, M.T., IJsselsteijn, W.A. and Heynderickx, I. "Visual discomfort in stereoscopic displays: a review". In "Stereoscopic Displays and Virtual Reality Systems XIV", pages 64900I–64900I. SPIE (2007). doi:10.1117/12.705527. 12

Lang, M., Hornung, A., Wang, O., Poulakos, S., Smolic, A. and Gross, M. "Nonlinear disparity mapping for stereoscopic 3D". In "SIGGRAPH", pages 75:1–75:10. ACM (2010). doi:10.1145/1833349.1778812. 42, 43, 44, 48, 50, 54, 139, 141, 147

Laveau, S. and Faugeras, O.D. "3D scene representation as a collection of images and fundamental matrices". Tech. Report 2205, INRIA (1994). 58, 84

Levin, A., Zomet, A., Peleg, S. and Weiss, Y. "Seamless image stitching in the gradient domain". In "European Conference on Computer Vision", pages 377–389. Springer (2004). doi:10.1007/978-3-540-24673-2_31. 111

Levoy, M. and Hanrahan, P. "Light field rendering". In "SIGGRAPH", pages 31–42. ACM (1996). doi:10.1145/237170.237199. 58, 62, 131

Lin, H.S., Guan, S.H., Lee, C.T. and Ouhyoung, M. "Stereoscopic 3D experience optimization using cropping and warping". In "SIGGRAPH Asia Sketches", pages 40:1–40:2. ACM (2011). doi:10.1145/2077378.2077428. 44

Lippman, A. "Movie-maps: An application of the optical videodisc to computer graphics". In "SIGGRAPH", pages 32–42. ACM (1980). doi:10.1145/800250.807465. 58

Lipski, C., Klose, F. and Magnor, M. "Correspondence and depth-image based rendering a hybrid approach for free-viewpoint video". In *Transactions on Circuits and Systems for Video Technology*, 24(6):942–951 (2014). doi:10.1109/TCSVT.2014.2302379. 60, 156

Lipski, C., Linz, C., Berger, K., Sellent, A. and Magnor, M. "Virtual video camera: Image-based viewpoint navigation through space and time". In *Computer Graphics Forum*, 29(8):2555–2568 (2010). doi:10.1111/j.1467-8659.2010.01824.x. 60, 156

Lipton, L. *Foundations of the stereoscopic cinema: a study in depth.* Van Nostrand Reinhold (1982). 7, 8, 9, 35

Liu, C.W., Huang, T.H., Chang, M.H., Lee, K.Y., Liang, C.K. and Chuang, Y.Y. "3D cinematography principles and their applications to

stereoscopic media processing". In "International Conference on Multimedia", pages 253–262. ACM (2011). doi:10.1145/2072298.2072332. 15

Liu, S. and Cooper, D. "Statistical inverse ray tracing for image-based 3D modeling". In *Transactions on Pattern Analysis and Machine Intelligence*, 36(10):2074–2088 (2014). doi:10.1109/TPAMI.2014.2315820. 65

Loop, C. and Zhang, Z. "Computing rectifying homographies for stereo vision". In "Conference on Computer Vision and Pattern Recognition", volume 1. IEEE (1999). doi:10.1109/CVPR.1999.786928. 20, 21, 22

Lowe, D.G. "Distinctive image features from scale-invariant keypoints". In *International Journal of Computer Vision*, 60(2):91–110 (2004). doi:10.1023/B:VISI.0000029664.99615.94. 42, 86

Mac Aodha, O., Brostow, G.J. and Pollefeys, M. "Segmenting video into classes of algorithm-suitability". In "Conference on Computer Vision and Pattern Recognition", pages 1054–1061. IEEE (2010). doi:10.1109/CVPR.2010.5540099. 65, 66

Mahajan, D., Huang, F.C., Matusik, W., Ramamoorthi, R. and Belhumeur, P. "Moving gradients: A path-based method for plausible image interpolation". In "SIGGRAPH", pages 42:1–42:11. ACM (2009). doi:10.1145/1576246.1531348. 59

Marr, D. and Poggio, T. "Cooperative computation of stereo disparity". In *Science*, 194(4262):283–287 (1976). 22

Masaoka, K., Hanazato, A., Emoto, M., Yamanoue, H., Nojiri, Y. and Okano, F. "Spatial distortion prediction system for stereoscopic images". In *Journal of Electronic Imaging*, 15(1):013002–013002 (2006). doi:10.1117/1.2181178. 26, 37, 38

Masia, B., Wetzstein, G., Aliaga, C., Raskar, R. and Gutierrez, D. "Display adaptive 3D content remapping". In *Computers & Graphics*, 37(8):983–996 (2013). doi:10.1016/j.cag.2013.06.004. 44

Matsuyama, T., Nobuhara, S., Takai, T. and Tung, T. *3D video and its applications*. Springer (2012). 158

Matthies, L., Kanade, T. and Szeliski, R. "Kalman filter-based algorithms for estimating depth from image sequences". In *International Journal of Computer Vision*, 3(3):209–238 (1989). doi:10.1007/BF00133032. 65

Matusik, W. and Pfister, H. "3D TV: A scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes". In "SIGGRAPH", pages 814–824. ACM (2004). doi:10.1145/1015706.1015805. 60

McCormack, J., Perry, R., Farkas, K.I. and Jouppi, N.P. "Feline: fast elliptical lines for anisotropic texture mapping". In "SIGGRAPH", pages 243–250. ACM Press/Addison-Wesley Publishing Co. (1999). doi:10.1145/311535.311562. 74

McMillan, L. and Bishop, G. "Plenoptic modeling: An image-based rendering system". In "SIGGRAPH", pages 39–46. ACM (1995). doi:10.1145/218380.218398. 58, 149

Mendiburu, B. *3D Movie Making: Stereoscopic Digital Cinema from Script to Screen*. Focal press (2009). 10, 13, 14, 15, 16, 23, 32, 33, 35, 49, 52, 159

Mendiburu, B. *3D TV and 3D cinema: tools and processes for creative stereoscopy*. Focal press (2011). 52, 159

Moezzi, S., Tai, L.C. and Gerard, P. "Virtual view generation for 3D digital video". In *Multimedia*, 4(1):18–26 (1997). doi:10.1109/93.580392. 60

Moulon, P., Monasse, P. and Marlet, R. "Adaptive structure from motion with a contrario model estimation". In "Asian Conference on Computer Vision", pages 257–270. Springer (2013). doi:10.1007/978-3-642-37447-0_20. 85, 93

Mumford, D. "The bayesian rationale for energy functionals". In *Geometry-driven diffusion in Computer Vision*, 1:141–153 (1994). 67

Neumann, R. "The lion king 3D: in-depth with disney". http://www.fxguide.com/featured/the-lion-king-3d-in-depth-with-disney/ (2011). [Online; accessed 14-August-2015]. 137, 138

Ng, K.C., Trivedi, M. and Ishiguro, H. "Generalized multiple baseline stereo and direct virtual view synthesis using range-space search, match, and render". In *International Journal of Computer Vision*, 47(1-3):131–147 (2002). doi:10.1023/A:1014589723611. 59, 65

Oh, K.J., Yea, S. and Ho, Y.S. "Hole filling method using depth based inpainting for view synthesis in free viewpoint television and 3D video". In "Picture Coding Symposium, 2009. PCS 2009", pages 1–4. IEEE (2009). doi:10.1109/PCS.2009.5167450. 43

Okutomi, M. and Kanade, T. "A multiple-baseline stereo". In *Transactions on Pattern Analysis and Machine Intelligence*, 15(4):353–363 (1993). doi:10.1109/34.206955. 23, 25, 64, 94

Oskam, T., Hornung, A., Bowles, H., Mitchell, K. and Gross, M. "OSCAM - optimized stereoscopic camera control for interactive 3D". In "SIGGRAPH Asia", pages 189:1–189:8. ACM (2011). doi:10.1145/2024156.2024223. 159

Phong, B.T. "Illumination for computer generated pictures". In *Communications of the ACM*, 18(6):311–317 (1975). doi:10.1145/360825.360839. 87

Pighin, F., Hecker, J., Lischinski, D., Szeliski, R. and Salesin, D.H. "Synthesizing realistic facial expressions from photographs". In "SIGGRAPH Courses", pages 75–84. ACM (1998). doi:10.1145/1198555.1198589. 62

Pinskiy, D., Longson, J., Kristof, P., Goldberg, E. and Neuman, R. "Stereo compositing accelerated by quadtree structures in piecewise linear and curvilinear spaces". In "Symposium on Digital Production", pages 13–20. ACM (2013). doi:10.1145/2491832.2491833. 53, 54, 149

Pitié, F., Baugh, G. and Helms, J. "Depthartist: a stereoscopic 3D conversion tool for CG animation". In "European Conference on Visual Media Production", pages 32–39. ACM (2012). doi:10.1145/2414688.2414693. 47, 49, 50, 139

Pollefeys, M., Nistér, D., Frahm, J.M., Akbarzadeh, A., Mordohai, P., Clipp, B., Engels, C., Gallup, D., Kim, S.J., Merrell, P. *et al.* "Detailed real-time urban 3D reconstruction from video". In *International Journal of Computer Vision*, 78(2-3):143–167 (2008). doi:10.1007/s11263-007-0086-4. 85

Prevoteau, J., Chalençon-Piotin, S., Debons, D., Lucas, L. and Remion, Y. "Multiview shooting geometry for multiscopic rendering with controlled distortion". In *International Journal of Digital Multimedia Broadcasting*, 2010 (2010). doi:10.1155/2010/975674. 151

Proferes, N.T. *Film Directing Fundamentals: see your film before shooting*. Focal press (2008). 159

Pujades, S., Boiron, L., Ronfard, R. and Devernay, F. "Dynamic stereoscopic previz". In "International Conference on 3D Imaging", pages 1–8. IEEE (2014). doi:10.1109/IC3D.2014.7032600. 38, 159

Pujades, S. and Devernay, F. "Viewpoint interpolation: Direct and variational methods". In "International Conference on Image Processing", pages 5407–5411. IEEE (2014). doi:10.1109/ICIP.2014.7026094. 107, 130

Pulli, K., Hoppe, H., Cohen, M., Shapiro, L., Duchamp, T. and Stuetzle, W. "View-based rendering: Visualizing real objects from scanned range and color data". In J. Dorsey and P. Slusallek, editors, "Rendering Techniques", Eurographics, pages 23–34. Springer (1997). doi:10.1007/978-3-7091-6858-5_3. 62

Raskar, R. and Low, K.L. "Blending multiple views". In "Pacific Conference on Computer Graphics and Applications", pages 145–153. IEEE (2002). doi:10.1109/PCCGA.2002.1167848. 63, 92, 111

Reynolds, M., Dobos, J., Peel, L., Weyrich, T. and Brostow, G.J. "Capturing time-of-flight data with confidence". In "Conference on Computer Vision and Pattern Recognition", pages 945–952. IEEE (2011). doi:10.1109/CVPR.2011.5995550. 65

Ronfard, R. and Taubin, G. "Introducing 3D Cinematography". In *Computer Graphics and Applications*, 27(3):18–20 (2007). doi:10.1109/MCG.2007.64. 2, 153

Ronfard, R. and Taubin, G. *Image and geometry processing for 3-D cinematography*, volume 5. Springer Verlag, Geometry and Computing (2010). 2, 153

Roth, S. and Black, M.J. "Fields of experts: A framework for learning image priors". In "Conference on Computer Vision and Pattern Recognition", volume 2, pages 860–867. IEEE (2005). doi:10.1109/CVPR.2005.160. 73, 112

Sattler, T., Leibe, B. and Kobbelt, L. "SCRAMSAC: Improving RANSAC's efficiency with a spatial consistency filter". In "International Conference on Computer Vision", pages 2090–2097. IEEE (2009). doi:10.1109/ICCV.2009.5459459. 42

Scharstein, D. and Szeliski, R. "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms". In *International Journal of Computer Vision*, 47(1-3):7–42 (2002). doi:10.1023/A:1014573219977. 40

Seitz, S.M., Curless, B., Diebel, J., Scharstein, D. and Szeliski, R. "A comparison and evaluation of multi-view stereo reconstruction algorithms". In "Conference on Computer Vision and Pattern Recognition", volume 1, pages 519–528. IEEE (2006). doi:10.1109/CVPR.2006.19. 64

Seitz, S.M. and Dyer, C.R. "Photorealistic scene reconstruction by voxel coloring". In *International Journal of Computer Vision*, 35(2):151–173 (1999). doi:10.1023/A:1008176507526. 64

Seuntiens, P., Meesters, L. and Ijsselsteijn, W. "Perceived quality of compressed stereoscopic images: Effects of symmetric and asymmetric jpeg coding and camera separation". In *Transactions on Applied Perception*, 3(2):95–109 (2006). doi:10.1145/1141897.1141899. 150

Shade, J., Gortler, S., He, L.W. and Szeliski, R. "Layered depth images". In "SIGGRAPH", pages 231–242. ACM (1998). doi:10.1145/280814.280882. 58

Shahrokni, A., Mei, C., Torr, P. and Reid, I. "From visual query to visual portrayal". In "British Machine Vision Conference", pages 117.1–117.10. BMVA Press (2008). doi:10.5244/C.22.117. 64

Shamir, A. and Sorkine, O. "Visual media retargeting". In "SIGGRAPH ASIA Courses", pages 11:1–11:13. ACM (2009). doi:10.1145/1665817.1665828. 43

Shan, Q., Jia, J. and Agarwala, A. "High-quality motion deblurring from a single image". In "SIGGRAPH", pages 73:1–73:10. ACM (2008). doi:10.1145/1399504.1360672. 73

Shibata, T., Kim, J., Hoffman, D.M. and Banks, M.S. "The zone of comfort: Predicting visual discomfort with stereo displays". In *Journal of vision*, 11(8):11 (2011). doi:10.1167/11.8.11. 12, 14, 15, 32, 47

Shum, H.Y., Chan, S.C. and Kang, S.B. *Image-based rendering.* Springer (2007). 55, 58, 98

Shum, H.Y. and He, L.W. "Rendering with concentric mosaics". In "SIGGRAPH", pages 299–306. ACM (1999). doi:10.1145/311535.311573. 58

Sinha, S.N., Scharstein, D. and Szeliski, R. "Efficient high-resolution stereo matching using local plane sweeps". In "Conference on Computer Vision and Pattern Recognition", pages 1582–1589. IEEE (2014). doi:10.1109/CVPR.2014.205. 42

Sinha, S.N., Steedly, D. and Szeliski, R. "Piecewise planar stereo for image-based rendering". In "International Conference on Computer Vision", pages 1881–1888. IEEE (2009). doi:10.1109/ICCV.2009.5459417. 65

Smolic, A., Kimata, H. and Vetro, A. "Development of MPEG standards for 3D and free viewpoint video". In "Three-Dimensional TV, Video, and Display IV", pages 60160R–60160R–12. SPIE (2005). doi:10.1117/12.631192. 61

Smolic, A. and McCutchen, D. "Report on 3DAV exploration of video-based rendering technology in MPEG". In *Transactions on Circuits and Systems for Video Technology*, 14(3):348–356 (2004). doi:10.1109/TCSVT.2004.823395. 60

Smolic, A., Muller, K., Dix, K., Merkle, P., Kauff, P. and Wiegand, T. "Intermediate view interpolation based on multiview video plus depth for advanced 3D video systems". In "International Conference on Image Processing", pages 2448–2451. IEEE (2008). doi:10.1109/ICIP.2008.4712288. 43, 59

SMPTE. "SMPTE STANDARD 196m-2003 motion-picture film - indoor theater and review room projection - screen luminance and viewing conditions". http://standards.smpte.org/content/st-196-2003/SEC1.body.pdf (2003). [Online; accessed 14-August-2015]. 36

SMPTE. "Society of motion picture & television engineers". http://www.smpte.org/ (2015). [Online; accessed 23-September-2015]. 36

Snavely, N., Garg, R., Seitz, S.M. and Szeliski, R. "Finding paths through the world's photos". In "SIGGRAPH", pages 15:1–15:11. ACM (2008). doi:10.1145/1399504.1360614. 59

Snavely, N., Seitz, S.M. and Szeliski, R. "Photo tourism: Exploring photo collections in 3D". In "SIGGRAPH", pages 835–846. ACM (2006). doi:10.1145/1141911.1141964. 59

Spottiswoode, R., Spottiswoode, N.L. and Smith, C. "Basic principles of the three-dimensional film". In *Journal of the Society of Motion Picture and*

Television Engineers, 59(4):249–286 (1952). doi:10.5594/J01778. 15, 26, 32, 35, 36, 49

Stich, T., Linz, C., Wallraven, C., Cunningham, D. and Magnor, M. "Perception-motivated interpolation of image sequences". In *Transactions on Applied Perception*, 8(2):11:1–11:25 (2011). doi:10.1145/1870076.1870079. 59

Strecha, C., von Hansen, W., Gool, L.V., Fua, P. and Thoennessen, U. "On benchmarking camera calibration and multi-view stereo for high resolution imagery". In "Conference on Computer Vision and Pattern Recognition", pages 1–8. IEEE (2008). doi:10.1109/CVPR.2008.4587706. 100

Sun, G. and Holliman, N. "Evaluating methods for controlling depth perception in stereoscopic cinematography". In "Stereoscopic Displays and Applications XX", pages 72370I–72370I–12. SPIE (2009). doi:10.1117/12.807136. 35

Szeliski, R. *Computer vision: algorithms and applications*. Springer (2010). xi, 18, 20, 21, 22, 25

Takahashi, K. "Theory of optimal view interpolation with depth inaccuracy". In "European Conference on Computer Vision: Part IV", pages 340–353. Springer (2010). 63, 64

Takahashi, K. and Naemura, T. "Super-resolved free-viewpoint image synthesis using semi-global depth estimation and depth-reliability-based regularization". In Y.S. Ho, editor, "Advances in Image and Video Technology", pages 22–35. Springer (2012). doi:10.1007/978-3-642-25367-6_3. 63, 92

Taneja, A., Ballan, L., Puwein, J., Brostow, G.J. and Pollefeys, M. "3D reconstruction and video-based rendering of casually captured videos". In D. Cremers, M. Magnor, M.R. Oswald and L. Zelnik-Manor, editors, "Video Processing and Computational Video", pages 77–103. Springer (2011). doi:10.1007/978-3-642-24870-2_4. 59

Tanimoto, M. "FTV (free-viewpoint television". In *Transactions on Signal and Information Processing*, 1(e4):454–461 (2012). doi:10.1017/ATSIP.2012.5. 60

THX. "THX". http://www.thx.com/ (2015). [Online; accessed 23-September-2015]. 36

THX. "THX certified cinema screen placement". http://www.thx.com/professional/cinema-certification/cinema-specifications/thx-certified-cinema-screen-placement/ (2015a). [Online; accessed 14-August-2015]. 36

THX. "THX HDTV setup". http://www.thx.com/consumer/home-entertainment/home-theater/hdtv-set-up/ (2015b). [Online; accessed 14-August-2015]. 36

Todd, J.T. "The visual perception of 3D shape". In *Trends in Cognitive Sciences*, 8(3):115 – 121 (2004). doi:10.1.1.89.5579. 7

Torralba, A. and Oliva, A. "Statistics of natural image categories". In *Network: computation in neural systems*, 14(3):391–412 (2003). 121

Triggs, B., McLauchlan, P.F., Hartley, R.I. and Fitzgibbon, A.W. "Bundle adjustment – a modern synthesis". In "International Workshop on Vision Algorithms: Theory and Practice", pages 298–372. Springer (2000). 85

Ukai, K. and Howarth, P.A. "Visual fatigue caused by viewing stereoscopic motion images: Background, theories, and observations". In *Displays*, 29(2):106–116 (2008). doi:10.1016/j.displa.2007.09.004. 12

Vaish, V. and Adams, A. "The (New) Stanford Light Field Archive". `http://lightfield.stanford.edu` (2008). [Online; accessed 14-August-2015]. 80

Vangorp, P., Chaurasia, G., Laffont, P.Y., Fleming, R.W. and Drettakis, G. "Perception of visual artifacts in image-based rendering of façades". In *Computer Graphics Forum*, 30(4):1241–1250 (2011). doi:0.1111/j.1467-8659.2011.01983.x. 63

Vedula, S., Baker, S. and Kanade, T. "Image-based spatio-temporal modeling and view interpolation of dynamic events". In *Transactions on Graphics*, 24(2):240–261 (2005). doi:10.1145/1061347.1061351. 58

Wang, Y.S., Tai, C.L., Sorkine, O. and Lee, T.Y. "Optimized scale-and-stretch for image resizing". In "SIGGRAPH Asia", pages 118:1–118:8. ACM (2008). doi:10.1145/1457515.1409071. 43

Wang, Z., Bovik, A.C., Sheikh, H.R. and Simoncelli, E.P. "Image quality assessment: from error visibility to structural similarity". In *Transactions on Image Processing*, 13(4):600–612 (2004). doi:10.1109/TIP.2003.819861. 80

Wanner, S. and Goldluecke, B. "Spatial and angular variational super-resolution of 4D light fields". In A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato and C. Schmid, editors, "European Conference on Computer Vision", pages 608–621. Springer (2012). doi:10.1007/978-3-642-33715-4_44. 4, 56, 57, 63, 67, 71, 72, 73, 76, 79, 80, 81, 83, 84, 100, 102, 103, 104, 105, 107, 108, 130, 131, 133, 135, 142, 146, 148, 154, 163, 164, 171, 172

Wanner, S. and Goldluecke, B. "Variational light field analysis for disparity estimation and super-resolution". In *Transactions on Pattern Analysis and Machine Intelligence*, 36(3):606–619 (2014). doi:10.1109/TPAMI.2013.147. 80

Wanner, S., Meister, S. and Goldluecke, B. "Datasets and benchmarks for densely sampled 4D light fields". In "Vision Modeling and Visualization", The Eurographics Association (2013). doi:10.2312/PE.VMV.VMV13.225-226. 80

Ward, B., Kang, S.B. and Bennett, E.P. "Depth director: A system for adding depth to movies". In *Computer Graphics and Applications*, 31(1):36–48 (2010). doi:10.1109/MCG.2010.103. 50

Werlberger, M., Trobin, W., Pock, T., Wedel, A., Cremers, D. and Bischof, H. "Anisotropic huber-L1 optical flow". In "British Machine Vision Conference", pages 108.1–108.11. BMVA Press (2009). doi:10.5244/C.23.108. 42

Wheatstone, C. "On some remarkable, and hitherto unobserved, phenomena of binocular vision". In *Philosophical Transactions of the Royal Society of London*, 128:371–394 (1838). doi:10.1098/rstl.1838.0019. 9

Wolberg, G. "Image morphing: a survey". In *The Visual Computer*, 14(8):360–372 (1998). doi:10.1007/s003710050148. 59

Woo, M., Neider, J., Davis, T. and Shreiner, D. *OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 1.2*. Addison-Wesley Longman Publishing Co., Inc., 3rd edition (1999). ISBN 0201604582. 143

Wood, D.N., Azuma, D.I., Aldinger, K., Curless, B., Duchamp, T., Salesin, D.H. and Stuetzle, W. "Surface light fields for 3D photography". In "SIGGRAPH", pages 287–296. ACM (2000). doi:10.1145/344779.344925. 62

Woods, A.J., Docherty, T. and Koch, R. "Image distortions in stereoscopic video systems". In "Stereoscopic Displays and Applications IV", volume 1915, pages 36–48. SPIE (1993). doi:10.1117/12.157041. 26

Yamanoue, H., Okui, M. and Okano, F. "Geometrical analysis of puppet-theater and cardboard effects in stereoscopic HDTV images". In *Transactions on Circuits and Systems for Video Technology*, 16(6):744–752 (2006). doi:10.1109/TCSVT.2006.875213. 11, 12, 34

Yan, T., Lau, R.W., Xu, Y. and Huang, L. "Depth mapping for stereoscopic videos". In *International Journal of Computer Vision*, 102(1-3):293–307 (2013). doi:10.1007/s11263-012-0593-9. 43, 44, 147

Yano, S., Emoto, M. and Mitsuhashi, T. "Two factors in visual fatigue caused by stereoscopic HDTV images". In *Displays*, 25(4):141–150 (2004). doi:10.1016/j.displa.2004.09.002. 15

Yeh, Y.Y. and Silverstein, L.D. "Limits of fusion and depth judgment in stereoscopic color displays". In *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 32(1):45–60 (1990). doi:10.1177/001872089003200104. 12

Zeisl, B., Georgel, P.F., Schweiger, F., Steinbach, E.G., Navab, N. and Munich, G. "Estimation of location uncertainty for scale invariant features points". In "British Machine Vision Conference", pages 57.1–57.12. BMVA Press (2009). doi:10.5244/C.23.57. 86

Zilly, F., Muller, M., Eisert, P. and Kauff, P. "The Stereoscopic Analyzer – An image-based assistance tool for stereo shooting and 3D production". In "International Conference on Image Processing", pages 4029–4032. IEEE (2010). doi:10.1109/ICIP.2010.5649828. 13

Zilly, F., Kluger, J. and Kauff, P. "Production rules for stereo acquisition". In *Proceedings of the IEEE*, 99(4):590–606 (2011). doi:10.1109/JPROC.2010.2095810. 13

Zinger, S., Do, L. and de With, P. "Free-viewpoint depth image based rendering". In *Journal of Visual Communication and Image Representation*, 21(5):533–541 (2010). doi:10.1016/j.jvcir.2010.01.004. 43, 61, 107

Zitnick, C.L., Kang, S.B., Uyttendaele, M., Winder, S. and Szeliski, R. "High-quality video view interpolation using a layered representation". In "SIGGRAPH", pages 600–608. ACM (2004). doi:10.1145/1186562.1015766. 60