PLENOPTIC LIGHT TRANSPORT

ADRIAN JARABO

SUPERVISOR: DIEGO GUTIERREZ

Tesis Doctoral - Ingeniería Informática
Departamento de Informática e Ingeniería de Sistemas
Universidad de Zaragoza

September 2015

To Domingo & Milagros, for the brain.

To Andrea, for the courage.

To Sofía, for the heart.

No rush to go home...

ABSTRACT

In this thesis we focus on the multidimensional nature of light transport as described by the plenoptic function, and in particular in the angular and temporal domains. While traditional imaging has been limited to bidimensional images, the emerging field of *Computational Imaging* has made increasingly available more complex multidimensional visual data, disambiguating additional domains of the plenoptic function. However, this higher dimensionality requires changing the way that visual information is processed, manipulated, visualized or synthesized. In this thesis we present contributions on these topics, addressing the challenges of adapting and rethinking them to handle higher-dimensional visual information. Specifically, within the *angular domain* we focus on light field editing, studying interaction paradigms and user workflows when interacting with light fields, and on spatio-angular filtering of complex appearances modeled with BTFs, studying how filtering affects appearance perception. On the other hand, in the *temporal domain* we focus on transient light transport, where the speed of light cannot longer be considered infinite, including contributions on capture and data processing, light transport simulation and visualization of time-resolved data.

MEASURABLE CONTRIBUTIONS

This thesis has led to the following results, which can be found in detail in Section 1.4:

- 6 JCR-indexed journal publications (3 of them ACM Transactions on Graphics) [99, 105, 224, 226, 98, 103]

- 5 peer-reviewed conference publications (including one SIGGRAPH Talk) [148, 100, 6, 225, 97]

- 1 peer-reviewed tutorial course [61]

- 1 research stay (three months) at Microsoft Research Asia in Beijing

- 1 research stay (five months) at Trinity College Dublin

- 1 research visit (eight days) at Tsinghua University

- 4 supervised PFC and 1 more in progress

- 1 best paper award, 1 paper selected in the *ACM Best of 2013 list*

- 5 invited talks

- Participation in 5 research projects

- Reviewer for 7 journals and 7 international conferences, and program committee member for 2 conferences

RESUMEN

Esta tesis se centra en la naturaleza multidimensional del transporte de luz, como describe la función plenóptica. Ésta modela la luz en un espacio heptadimensional, en función de la posición, dirección, longitud de onda (color) y tiempo. La imagen tradicional, incluyendo fotografía o imagen sintética, proyecta este espacio multidimensional en el plano bidimensional del sensor de la cámara, integrando las dimensiones angular y temporal. Esto resulta en una pérdida significativa de información, reduciendo las capacidades de extracción de información visual, y de manipulación de la misma.

Con la aparición del campo de *Imagen Computacional (Computational Imaging)* la información contenida en las dimensiones integradas en el sensor, antes perdida, es ahora recuperable. Esta información adicional abre un gran número de aplicaciones en p.ej. visión por computador o imagen médica. Sin embargo, este incremento en la dimensionalidad obliga a cambiar la forma en que los datos son procesados y manipulados, así como en su visualización o síntesis. Esta tesis presenta contribuciones en estas áreas, centrándose en desarrollar nuevas técnicas de manipulación, visualización y síntesis del transporte de luz incorporando las dimensiones *angular* y *temporal* de la función plenóptica. El objetivo es llenar el hueco existente entre la imagen tradicional bidimensional, en la que se ha centrado la mayoría de trabajo previo, y la nueva imagen plenóptica, de naturaleza multidimensional.

La tesis está dividida en dos partes: En la primera, nos centramos en el dominio angular, y en particular en las diferentes representaciones direccionales de apariencia. Primero presentamos un estudio sobre la interacción de usuario a la hora de editar *light fields*, con foco en preferencias y *workflows*, y proponemos una interfaz de edición de *light fields* basada en nuestros resultados. En el segundo trabajo dentro del dominio angular analizamos los efectos del filtrado en la percepción de la apariencia de materiales complejos modelados con *bidirectional texture functions* (BTFs). Basados en las conclusiones de nuestro análisis proponemos una serie de aplicaciones en compresión, filtrado y visualización eficiente de dichos materiales.

La segunda parte de la tesis se dedica al dominio temporal, y en particular al transporte de luz en estado transitorio, con contribuciones en el procesado, generación y visualización de datos visuales resueltos en el tiempo. Primero nos centramos en el procesado y visualización de imagen ultrarrápida, obtenida mediante un sistema capaz de capturar a una resolución efectiva de picosegundos. Después, presentamos un entorno de simulación transitoria de la iluminación, proponiendo diferentes estrategias de reconstrucción y muestreo adaptadas a las particularidades del transporte de luz resuelto en el tiempo. Finalmente, se describe un sistema de visualización interactiva de este tipo de datos, modelando los efectos relativistas resultantes del movimiento ultrarrápido de la cámara virtual.

Como resultado de esta tesis se han publicado 6 artículos en revista (incluyendo 3 en *ACM Transactions on Graphics*), 5 artículos y 1 tutorial en conferencias con revisión por pares, y se han realizado 2 estancias de investigación (8 meses en total).

ACKNOWLEDGEMENTS

Many people have helped and supported me during my life and academic career, and are in part responsible for the completion of this thesis. I owe them a debt of gratitude. I would like to thank a list of people in particular. Note that unfortunately this list is not exhaustive: those who have been left out, thank you.

Diego, for advising and teaching me to be a better researcher. For the hard work. For giving me the chance to do stuff I wanted to, trusting my judgment. And for the perspective, for showing me how to see the big picture instead of focus on just the details.

The members of the *Graphics & Imaging Lab*, for being an awesome group for discussing crazy ideas, for the lots of fun and work. For helping on deadlines, and for teaching me science, art or whatever. From them, I would like to thank in particular to *Adolfo Muñoz*, for teaching me tons of stuff during coffee breaks, and *Belen Masia*, for all the hard work, knowledge and common sense put in all projects we have work together, which are a signficant part of this thesis.

The students I have supervised, *Carlos Aliaga*, *Julio Marco*, *Raúl Buisán*, *Ibón Guillén* and *David Guimerá*. Thanks for letting me experiment different research ideas with you, trusting, maybe thoughtlessly, in my knowledge and experience. I hope you have learned something, or at least, that you have had fun: that is the main reason to do what we do.

The collaborators that I have had during these years. It is a long list. Thanks for sharing your deep knowledge, for putting on the table different points of view, and for working so hard making our research projects successful.

My hosts at Dublin and Beijing, *Carol O'Sullivan* and *Steve Lin*. Thanks for welcoming and hosting me during my months there, for letting me be part of your groups. I learned many things there.

The anonymous reviewers for their disinterested effort on improving our papers: while rejections always hurt, they have made our work better. And thanks also to all the giants in whose shoulders this work stands, for all these awesome papers that inspired my work.

My family and friends, for their support, for their interest in my work, even when it is not very clear what I do. For their love, even though I am an awful grandson, nephew, cousin or friend. For still including me in their plans. They deserve both an acknowledgement and an apology.

My parents, Domingo and Milagros. For their love, care and education. For teaching me to work hard, and encouraging me to do what I like. For showing me that learning is awesome. For being the first on telling me to get a PhD, and the first on rushing me to finish it. For having made all what I am, both genetic- and education-wise.

My sister Andrea. For growing with me. For making me compete, trying hard to be better than her. Of course I have miserably failed on that, but if I am any good in something is mostly her fault.

and *Sofía*. She has suffered this thesis the most, and she deserves all forms of acknowledgment. Thanks for sharing with me these years, for the support and help. For the enthusiasm when being showed renders. For teaching me that resting increases productivity, and forgiving the crunching at deadlines. For watching sunsets with me, even though I would be probably thinking on the scattering phenomena. I have no idea why you are still with me, but could not be more grateful.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Part I

INTRODUCTION & OVERVIEW

INTRODUCTION

Light, has always intrigued the human being. Since ancient times several theories have tryed to explain why do we see, and *what is light*. This is not surprising: Visual data is one of the main sources of information on how the world around us is. It gives information on the shape and material of objects which is crucial for their recognition, and is the main information used for navigation around the world. In fact, it is the sense with longer sensibility, allowing us to see up to distances of 2.5 million light years with the naked eye (the distance to the Andromeda galaxy).

Moreover, light does not only play a fundamental role on survival. Its emission and interaction with matter creates visual effects of great beauty, from massive galaxies and nebulae to rainbows, auroras or, at very small scale, the iridescent wings of butterflies. Such visual richness has inspired painters and photographers, but also scientists aiming understand the nature of light and why objects look as they look, and looking for ways to measure and capture it.

In this thesis we focus on the multidimensional nature of light transport as described by the plenoptic function, and in particular in the angular and temporal domains, motivated by the emerging field of *Computational Imaging*. This multidimensionality changes the way that visual information, traditionally limited to bidimensional images, is processed, manipulated, visualized or synthesized. In this thesis we present contributions on these topics, addressing the challenges of adapting and rethinking them to handle higher-dimensional visual information. In the following we provide a brief overview of light transport and imaging from the plenoptic perspective, and summarize the contributions of this thesis.

1.1 THE PLENOPTIC FUNCTION

The plenoptic function [2] describes the visual information in a scene in any point in space and time. It models the incoming wavelength-dependent radiance L_λ with wavelength (color) λ at a point $\mathbf{x} = (\mathbf{x}_x, \mathbf{x}_y, \mathbf{x}_z)$ from a particular direction $\omega = (\phi, \theta)$ at a given time t, as a seven dimension function

$$\mathscr{P}(\mathbf{x}_x, \mathbf{x}_y, \mathbf{x}_z, \phi, \theta, \lambda, t) = L_\lambda. \tag{1.1}$$

Intuitively, the space defined by $\mathscr{P}$ models *all possible photographs that can ever be taken*. Of course, this function is a simplification of the actual physics of light, since it ignores the wave (e.g. polarization or phase) or quantum properties of light. Nevertheless it is still valuable to reason about light transport in general, and imaging in particular.

Traditional imaging samples this space by taking only a slice of $\mathscr{P}$ on the bidimensional sensor. This makes the formation of the 2D image $\mathscr{I}(\mathbf{x}, \lambda)$ an integration along the angular and temporal domains[1]

$$\mathscr{I}(\mathbf{x}, \lambda) = \iint \mathscr{P}(\mathbf{x}, \omega, \lambda, t)\, \mathrm{d}\omega\, \mathrm{d}t. \tag{1.2}$$

This form of imaging is influenced by how the the human eye captures images, where light enters through the cornea and the photo-receptors (cones and rods) in the retina integrate the incoming angular domain along a certain exposure time (the temporal sensibility of the eye is over 10 miliseconds depending on the type of signal [231]). Of course, capturing and representing the visual information as a bidimensional image $\mathscr{I}(\mathbf{x}, \lambda)$ has dramatically influenced the way we manipulate and generate visual content. First of all, traditional displays were developed to show bidimensional images, being optimized to match as much as possible the capabilities of the human eye (in terms of e.g. frame-rate or wavelength). Additionally, realistic image synthesis focuses on reconstructing the captured image as if it were captured with a virtual camera [124, 73] or even the human eye [188], with significant efford on accurately computing the double integral in angle and time in Equation (1.2) responsible of effects such as defocus or motion blur [260]. Finally, the combination of *bidimensional* imaging, synthesis and display has made that both manipulation and interaction with visual data had been optimized to this type of content, including the interaction workflows and tools, as can be observed in commercial image manipulation software such as Photoshop[TM].

However, integrating over different domains in Equation (1.2) removes the information contained in these domains, which cannot be recovered back. This significantly limits the information that can obtained from the scene through a photograph. While this can be partially recovered by a human observer, it is very hard for a computer to interpret the information encoded in a single bidimensional slice of the plenoptic function. Moreover, this strong dimensionality reduction through projection and integration also reduces the expressibility and manipulation capabilities of the captured visual data.

COMPUTATIONAL PHOTOGRAPHY   Recently it has emerged the field of *Computational Imaging* [185, 61], at the intersection between computer vision, optics and electronics, with the goal of avoiding the lost of information in traditional photography and therefore capturing more data from the scene. The main idea is to acquire more information of the scene by coding or multiplex different steps of the pipeline (e.g. lens, sensor, illumination...) to enhance the captured data, and then use this captured information to reconstruct the final image *after* the capture by means of computation. These new imaging techniques have opened up several new applications over visual data, including light transport capture and analysis [199, 173, 169], changing the camera setup in a post-capture basis [57, 94] or separating illumination components [164, 46].

More in particular, several works have been proposed to capture the different dimensions of the plenoptic function. The excellent survey by Wetzstein et al. [241] provides a thorough discussion on the different proposed methods. These works allow accessing the information carried by the dimensions

---

1 Note that strictly speaking there is also an integration along the spatial domain in the footprint of the pixel, and in the wavelength domain due transforming the full spectra into an RGB-value. Additionally, the dimensionality of the spatial domain is also reduced from $\mathbb{R}^3$ to $\mathbb{R}^2$ once the image is projected in the bidimensional sensor, taking only the x and y components of $\mathbf{x}$.
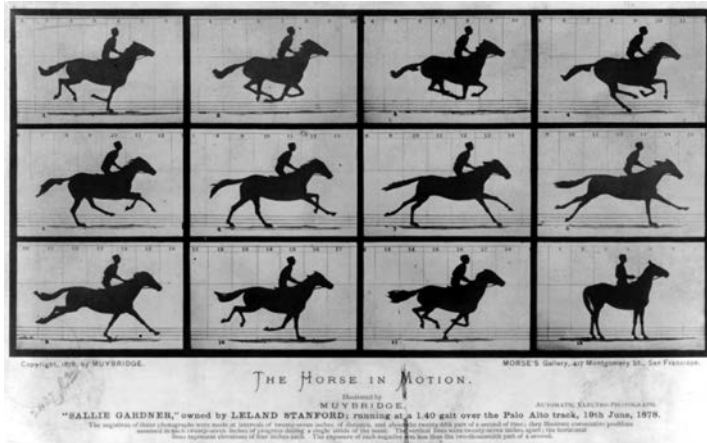
Figure 1.1: *The Horse in Motion* by Eadweard Muybridge (1878). To capture this sequence, Muybridge set a dozen cameras in line linked with a set of wires placed on the floor. These wires then triggered each camera as the horse ran past. Each capture had an exposure time of less than 2 miliseconds. Via Wikimedia Commons.

in the plenoptic function traditionally integrated during imaging, and therefore lost. This additional information has permited unprecedent capabilities for scene understanding and reconstruction from visual data, more powerful *image* processing algorithms or richer visualization of visual data. But it comes with the price that all previous knowledge on processing, manipulation, visualization or synthesis need to be rethough to be adapted to the multidimensional visual information now accessible. In this thesis we deal with these new challenges, presenting contributions on each of these for multidimensional plenoptic data, in particular for the two domains of integration in Equation (1.2): angle and time.

1.2   ANGULAR AND TEMPORAL DOMAINS

In this thesis we focus on the *angular* and *temporal* domains of the plenoptic function. These have been in general undersampled or integrated in traditional imaging with the aforementioned loss of information, and have recently gained significant attention since they allow extracting more information about the world while extending the capabilities of traditional photography. This does not mean that imaging these dimensions is new: the idea of sampling the *angular domain* of the plenoptic function, capturing the same scene from slightly different points of view was first proposed by Gabriel Lippmann in 1908 [139, 138] in what was termed *integral photography*. While this technique is fundamentally the origin of most following work on light field imaging [136, 51], and has also inspired the development of a variety of display systems, it is now when these devices are available in the consumer market in the form of light field cameras (e.g. Lytro[TM], Raytrix[TM] or PCam[TM]). This new kind of imaging allows sophisticated post-capture effects, such as view-panning [136, 56], refocus [92, 165], super-resolution [144] or 3D reconstruction [117], while improving robustness in computer vision in challenging conditions [26]. Moreover, light fields are the input for automulticopic displays [150], which allow to show stereo content to multiple viewers without the need of glasses or other equipment,

and that are currently available on TV screens, desktop monitors or even portable devices (e.g. consoles or mobile phones).

Similarly, sampling the *temporal domain* dates from more than a century: the first high-speed photography was captured by Eadweard Muybridge in the late 1800's, and was used to study the motion of a galloping horse by sequentially shooting a set of twelve cameras [154] (Figure 1.1). Since then, a number of capture devices were developed, from Edison's Kintergraph and Kintescope to the Lumières' Cinematographe that gave birth to the cinema [143]. These techniques aimed to captured videos, but many other research works focused on providing faster and faster exposure times; of particular interest is Harold Edgerton's stroboscope [35], developed in the mid 1900's, which allowed capturing stunning photographs of fast events where the time seems to freeze. While these techniques helped understanding mechanical behavior in fast events such as liquids or explosions, they were still unable to provide information to reason about the properties of the scene. This has changed with the appearance of *transient imaging*, which aims to capture at a temporal resolution enough to see the propagation of light. This field emerged with the *femto-photography* [225, 224] system, presented in this thesis in Chapter 4, which allows capturing videos at a temporal resolution in the order of picoseconds. Such a temporal resolution has opened-up new avenues of scene understanding and reconstruction such as non line-of-sight imaging [223, 79], capturing material properties [162, 161] or separating illumination components [245, 88]. In addition, while the original system was very expensive, new lower-cost systems have appear [78, 111] that tradeoff temporal resolution to allow faster and easier-to-use systems, that can even work in domestic time-of-flight cameras such as the Kinect^{TM} 2 device [161].

1.3   GOAL & OVERVIEW

The main goal of this thesis is to develop new techniques on processing, manipulation, visualization and synthesis of multidimensional visual data including the angular and the temporal domains of the plenoptic function. The aim is to fill the existing gap between traditional imaging, that takes only a spatial slice of the plenoptic function and where several works have addressed these challenges, and new plenoptic imaging, in particular focusing on the angular and temporal domains. This is of fundamental importance to the wide-spread of this multidimensional data in order to substitute, or at least become an alternative to, the traditional bi-dimensional images. This is the case of, for example, new editing algorithms developed to address the special characteristics of angularly-resolved data, which with the emergence of plenoptic cameras and automultiscopic displays might become a standard in a few years; or new computational techniques to effectively visualize and simulate time-resolved light transport, fundamental to devise new techniques for scene reconstruction.

OVERVIEW   This thesis is divided in two main parts, one for each of the two dimensions of interest of the plenoptic function:

- PART II deals with the angular domain of the plenoptic function, with focus on different visual representations of appearance. Chapter 2 is devoted to light fields [136, 56], and in particular with interaction and manipulation of this angularly-resolved visual data. We study different interaction paradigms for light fields and analyze the best way of editing them. In Chapter 3 we move our focus to bidirectional texture

functions (BTF) [25], that extends light fields including the directional domain of the incoming light by representing the full *reflectance field*. We analyze the effect of filtering on the perception of complex appearance modeled with BTFs, and propose a set of applications for our findings.

- PART III is devoted to the temporal domain, and in particular to ultrafast, time-resolved, light transport. In Chapter 4 we first introduce a system capable to capture at picosecond resolution, and the data processing necessary for correct visualization. This project was lead by colleagues at MIT Media Lab. Then, Chapter 5 focuses on time-resolved rendering by presenting a framework for effective transient rendering. We introduce a set of reconstruction and sampling techniques, together with simulations of non-trivial phenomena in transient state. Finally, Chapter 6 further explores the visualization of time-resolved data, taking into account the relativistic effects arising due to ultrafast camera motion through the scene.

While I am the leading author in most of the works presented here, they have been done in collaboration with different colleagues. Thus, at the beginning of each chapter the work described is put in context, and my contribution is explicitly described when needed.

1.4 CONTRIBUTIONS AND MEASURABLE RESULTS

1.4.1 *Publications*

Most of the work presented in this thesis has been already published, in particular in six journals indexed in JCR, including three papers in ACM Transactions on Graphics and presented at SIGGRAPH or SIGGRAPH Asia, and five peer-reviewed international conferences:

- Evaluation of Interaction Paradigms for Light Field Editing (Chapter 2, Part II):
  - The main work on light field editing was accepted in SIGGRAPH 2014, and published in ACM Transactions on Graphics [99]. This journal has an impact factor of 4.096, and its position in the JCR index is 1st out of 104 (Q1) in the category Computer Science, Software Engineering (data from 2014).
  - Further analysis on the workflows of the users when editing light fields (Section 2.6) has been published in the International Conference on Computer Graphics, Visualization, Computer Vision and Image Processing (CGVCVIP) 2014 [148].
  - Two papers describing preliminary results on edit propagation in light fields (Section 2.7) have been published in the Ibero-American Symposium in Computer Graphics (SIACG) 2011 [100] and in the Pacific Rim Conference on Multimedia [6] respectively.

- Effects of Filtering on the Appearance of BTF (Chapter 3, Part II):
  - This work has been published in IEEE Transactions on Visualization and Computer Graphics [105], and presented in Pacific Graphics 2014.. This journal has an impact factor of 2.482, and its position in the JCR index is 8th out of 104 (Q1) in the category Computer Science, Software Engineering (data from 2014).

- Femto-Photography (Chapter 4, Part III):
  - This work has been accepted in SIGGRAPH 2013, and published in ACM Transactions on Graphics [224]. This journal has an impact factor of 4.096, and its position in the JCR index is 1st out of 104 (Q1) in the category Computer Science, Software Engineering (data from 2014).
  - This work has been invited to be published as a *Research Highlight* in the journal *Communications of the ACM* [226]. This journal has an impact factor of 3.609, and its position in the JCR index is 5th out of 102 (Q1) in the category Computer Science, Theory & Methods, and 2nd out of 104 (Q1) in the category Computer Science, Software Engineering (data from 2014).
  - Previously, it was accepted as a talk to SIGGRAPH 2012 [225].

- A Framework for Transient Rendering (Chapter 5, Part III):
  - This work has been accepted in SIGGRAPH 2013, and published in ACM Transactions on Graphics [98]. This journal has an impact factor of 4.096, and its position in the JCR index is 1st out of 104 (Q1) in the category Computer Science, Software Engineering (data from 2014).

- Relativistic Effects for Time-Resolved Light Transport (Chapter 6, Part III):
  - This work has been published in Computer Graphics Forum [103], and presented in the Eurographics Symposium on Rendering (EGSR) 2015. This journal has an impact factor of 1.902, and its position in the JCR index is 18th out of 104 (Q1) in the category Computer Science, Software Engineering (data from 2014).
  - Previous results were published in CEIG 2013 [102].

In addition to these previous publications, during my PhD I have collaborated in other research projects not directly related with the topic of this thesis:

- Evaluation of the Effect of Shape and Material Stylization on the Perception of Computer Generated Faces.

  In this work, lead by Eduard Zell, we evaluate the effect that different artistic stylizations have on the perception of faces. It has been accepted in SIGGRAPH Asia 2015, and published in ACM Transactions on Graphics [256].

- Separable Subsurface Scattering.

  In this work, lead by Jorge Jimenez, we propose a method for high-quality simulation of subsurface scattering at real-time frame rates. It was published in Computer Graphics Forum [110], and presented in EGSR 2015. This technique is currently the state-of-the-art on real-time subsurface scattering, and had an enormous impact on the industry, being used by game companies such as Activision-Blizzard.

- A Biophysically-Based Model for Skin Aging.

  In this work, lead by Jose Iglesias, we propose a new skin model including the effects of aging, based on biophysical data. It was accepted in Eurographics 2015, and published in Computer Graphics Forum [90].

- Evaluation of the Perceived Fidelity of Illuminated Dynamic Scenes.

    In this work, we analyzed the effect on approximated global illumination on the quality of complex dynamic scenes. It was accepted in Eurographics 2012, and published in Computer Graphics Forum [104].

1.4.2 *Awards*

We include here a list of awards and fellowships received throughout this thesis, that have allowed the realization of the work here presented:

- FPI Grant from the Universidad de Zaragoza (4-year PhD grant).

- NVIDIA Academic Program: Tegra prototype gift (Mobile Computational Photography: Appearance Capture and Editing; with Belen Masia and Diego Gutierrez).

- CAI Europa Grant: Funding for research visit at Trinity College Dublin.

Additionally, some projects described in this thesis have been received different awards or recognitions:

- The paper *Femto-Photography: Capturing and Visualizing the Propagation of Light* was selected into the *ACM Best of 2013 list*.

- Best paper (1 in 2) at CEIG 2013 for the work *Rendering Relativistic Effects in Transient Imaging* (proposed for extension and submission to the journal Computer Graphics Forum; the extension got accepted to the journal).

1.4.3 *Research Stays and Visits*

Two research visits, totaling 8 months, were carried out during this PhD in two different institutions:

- February 2011 – June 2011 (five months): Visiting student at the *Graphics, Vision and Visualisation (GV2)* group, Trinity College Dublin. Supervisor: Prof. Dr. Carol O'Sullivan. Our work on perception of illumination on complex dynamic scenes [104] was the result of this stay.

- June 2013 – August 2013 (three months): Research intern at the *Internet Graphics Group*, Microsoft Research Asia. Supervisor: Dr. Steve Lin.

Additionally, a visit of 8 days to Tsinghua University (Beijing, China) took place in September 2014.

1.4.4 *Supervised Students*

During the development of this thesis I have supervised the Graduate Thesis of five students:

- Ongoing: David Guimera. *Physically-Accurate Light Transport in the Atmosphere*. Expected graduation date: February 2016.

- 2014: Ibon Guillen. *Progressive Instant Radiosity*. Grade: 9/10.

- 2014: Raul Buisan. *Bidirectional Clustering in Point-Based Global Illumination*. Grade: 9.3/10.

- 2013: Julio Marco. *Transient Light Transport in Participating Media*. Grade: 10/10.

- 2012: Carlos Aliaga. *Prefiltered global illumination in objects with complex geometry and reflectance*. Grade: 9/10.

1.4.5 *Research Projects*

During my PhD studies I have participated in the following research projects:

- VERVE: Vanquishing fear and apathy through e-inclusion: personalised and populated realistic virtual environments for clinical, home and mobile platforms. European Commission (FP7-ICT-2011-7). Grant no.: 288914. PI (in Spain): Diego Gutierrez.

- GOLEM: Realistic Virtual Humans. European Commission Marie Curie Industry–Academia Program, Seventh Framework. Grant no.: 251415. PI: Diego Gutierrez.

- LIGHTSLICE: Captura, análisis y aplicaciones del transporte de luz multidimensional (Aplicación a imagen médica). Spanish Ministry of Economy and Competitivity. PI: Diego Gutierrez.

- MIMESIS: Técnicas de bajo coste para la adquisición de modelos de apariencia de materiales. Spanish Ministry of Science and Education (TIN2010-21543). PI: Diego Gutierrez

- Femto-Fotografía: Análisis y Aplicaciones Prácticas. BBVA Foundation. PI: Diego Gutierrez.

Part II

ANGULAR DOMAIN

In this part we focus on the *angular domain* of the plenoptic function, and in particular on angularly-resolved representations of appearance. These representations include *light fields* and *bidirectional texture functions (BTF)*. We start focusing on editing and manipulation of light fields, describing our work on design and analysis of light field editing interfaces, with focus on preferences and workflows. Then, we move our focus to the second directionally-varying representation (BTFs), and analyze the perceptual effects of filtering this type of multidimensional data.

2

EVALUATION OF INTERACTION PARADIGMS FOR LIGHT FIELD EDITING

Here we describe a thorough study to evaluate different light field editing interfaces, tools and workflows from a user perspective. This is of special relevance given the arrival in the consumer market of light field cameras positioning light fields as an alternative for traditional photography, and the multidimensional nature of light fields, which may make common image editing tasks become complex in light field space. We propose an interface for light field editing based on the two main paradigms in the literature, and perform two different experiments, collecting both objective and subjective data from a varied number of point-based editing tasks of increasing complexity, in both real and synthetic light fields. I shared the first authorship of this work with Belen Masia, who was in charge of the analysis of the experiment, while I focused on the design and development of the editing interfaces, and the experimental design.

This work has been published in *ACM Transactions on Graphics* and presented at *SIGGRAPH 2014*. A follow up work analyzing in more depth the user workflows was later published in the *International Conference on Computer Graphics, Visualization, Computer Vision and Image Processing (CGVCVIP) 2014*. Additionally, we explored other types of interaction when editing light fields, in particular sparse edits propagation, as described in Section 2.7. While this is still research in progress, some initial results have been already published [100, 6].

A. Jarabo, B. Masia, A. Bousseau, F. Pellacini & D. Gutierrez
How Do People Edit Light Fields?
ACM. Trans. Graph. Vol.33(4), SIGGRAPH 2014

B. Masia, A. Jarabo & D. Gutierrez
Favored Workflows in Light Field Editing
Proc. of CGVCVIP 2014

2.1 INTRODUCTION

Light fields [136, 56] are rapidly gaining popularity as an alternative to digital photographs. Consumer light field cameras already exist (such as Raytrix^{TM} or Lytro^{TM}, even compact enough to be included in mobile devices [227]. As the number of captured and shared light fields increases, the need for editing tools arises as well. However, current light field editing techniques are limited to changing perspective or focus, or to applying some pre-defined filters [145, 228]. As opposed to the well-established editing of 2D images, user interfaces to edit light fields remain largely unexplored.

Editing light fields is a challenging task for several reasons. First, a light field is a four-dimensional data structure while the majority of displays and input devices today are designed for two-dimensional content. Second, light fields are redundant which implies that any local edit on a light field needs to be propagated coherently to preserve this redundancy. Finally, while light fields provide a vivid sense of depth, this depth information is not encoded
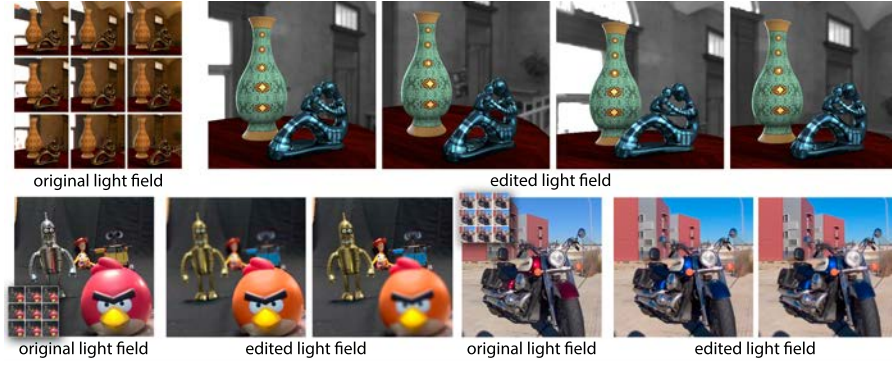
Figure 2.1: Example results of light fields edited by different users. Top: A synthetic light field (*vase*), with ground truth depth information. Bottom: Two real light fields (*toys* and *motorbike*) captured with the Lytro camera. In this work we evaluate the benefits of different light field interaction paradigms and tools, and draw conclusions to help guide future interface designs for light field editing.

explicitly. Light field user interfaces must take these properties into account to present the visual information in a legible way and to minimize redundant work for the user.

Current literature proposes two general paradigms to interact with light fields on two-dimensional devices. *Multiview* [258, 201, 229] relies on parallax to convey disparity. Interaction is constrained to the angle of view: Users can change the viewpoint, equivalent to limited displacement of a virtual camera. *Focus* interfaces [27], on the other hand, rely on depth-of-field blur. Interaction is constrained to the focus plane, which the users can change. However, neither of these have been thoroughly analyzed or validated.

In this work, our goal is to explore light field editing interfaces from a user perspective, and to provide a quantitative and qualitative evaluation of the effectiveness of different approaches. To do this, we build interfaces based on the two main existing light field editing paradigms, conduct a wide range of experiments with novice users, and analyze their performance and preferences. The experiments grow in complexity, and are divided in two studies. In the first one, we ask participants to perform tasks with both interfaces, based on simple edits and using synthetic light fields to be able to use accurate depth information. From the gained insights, we design a third interface, which additionally incorporates a series of new tools. This new interface and tools are used in our second study, which covers more advanced tasks (e.g. changing an object's appearance, or colorization). Moreover, this second study uses real light fields as input, which means that the recovered depth is not perfect. In designing the experiments, we seek to answer questions such as: Can users edit light fields, similar to common image editing? How do people interact with the 4D structure of a light field? What is the best method to locate a position in such 4D space? What is the preferred workflow for editing? Is the currently imperfect depth in real light fields good enough for editing? Does it affect performance?

CONCLUSIONS   The analysis of the collected objective and subjective data allows us to draw the following conclusions:

- novice users can edit captured light fields of different nature, although with varying workflows;

Figure 2.2: User interfaces used in our tests. Left: *multiview* paradigm. In it, windows are inter-changeable, and show two different views of the light field; epipolar lines (white) mark the trajectory a currently active stroke will follow when moved in depth. Right: *focus* paradigm, in which the in-focus plane marks the depth at which strokes will be placed; the window on the right shows the *un-blended* strokes (please see Section 2.3 for details).

- users rely on depth information frequently, finding it essential in many tasks;
- inaccuracies in depth information do not hinder users' performance when editing;
- a *multiview* approach is essential for navigating and viewing the light field edits;
- on the other hand, a *focus* approach is attractive to users because of the feeling of control it conveys; alternative tools, however, can also provide that feeling of control without focus;
- we also present tools with which users exploit the extra dimensionality of light fields and which, in combination with the editing paradigms, support artistic exploration.

LIMITATIONS AND SCOPE   The findings of our work can only be conclusive to the interfaces, tools and depth reconstruction methods used in the experiments. While the interfaces are the two most common interfaces found in previous literature on light field manipulation, other types of interfaces could be proposed and studied. We limit our study to point-based tools, but they represent common edits in most image editing software; additionally, point-based interaction represents the building block for more complex editing tools, such as selection or local filtering. Finally, our analysis on the influence of errors in depth is limited to the real depth maps reconstructed with three state-of-the-art depth reconstruction methods. However, despite these issues, we believe our work provides a solid ground for the design of light field editing interfaces, a reference for future researchers and interface designers. Thus, to further validate our methods and allow others to build on our work, we make our interfaces, raw data and analysis available online[1].

2.2 RELATED WORK

Different interfaces and interaction paradigms have been explored in the fields of computer graphics and human-computer interaction, for instance for lighting [115, 170], material editing [116, 22], video editing [194, 54] or painting [70]. In this work, we perform the first study focusing on the particular topic of light field editing.

---

1 http://giga.cps.unizar.es/~ajarabo/pubs/lfeiSIG14/

LIGHT FIELD EDITING   Most light field editing systems perform consistent operations over multiple views. Seitz and Kutulakos [197] estimate a voxel-based representation of a light field to propagate local edits, such as painting and scissoring, between multiple views of a scene. Jarabo et al. [100] propagate sparse edits in a light field based on pixel affinity. Related methods estimate depth in a stereo pair to perform consistent painting and copy-/pasting [206, 141, 178, 118]. While depth estimation assumes static scenes, other approaches rely on feature matching to propagate edits over image collections containing deformable objects [74, 69, 254].

In contrast, other systems require user intervention to indicate correspondences between two or more views. Zhang et al. [258] and subsequent work by Wang et al. [229] morph between two light fields by first requiring users to position polygons in several views, constrained by epipolar geometry. Users then indicate corresponding polygons in a second light field to guide the morph. In Pop-Up Light Field [201], users segment the light field into multiple depth layers by adjusting a polygon around the silhouette of each object in multiple views. Chen et al. [20] segment the light field into multiple volumes defined by the user, which can be deformed, while Horn and Chen [82] present a shader-like language to edit and compose light fields.

There are, additionally, a few prototype displays that allow light field editing using gesture tracking [147] or 3D light pen [213]. However, we choose to focus on interfaces that can be available to a wide audience without the need for what today still is specialized hardware. We further reject alternatives such as editing the light field in its epipolar volume form [117], or in a stereo interface, based on a pilot study.

The goal of these works is to develop specific editing tools, or to propagate edits consistently across views, but none of them provide an analysis of the interface itself. In contrast, we focus on analyzing how the user interacts with the four-dimensional structure of a light field in order to specify common editing operations, with the final goal of learning insights to design a light field editing interface.

2.3   OVERVIEW

INTERFACES   In this work, we first analyze two basic interfaces, based on interaction paradigms found in the light field editing literature. In one interface, which we call *multiview*, users navigate between the views of the light field to specify correspondences that locate their edits in space. This class of interfaces is the most common in existing work on non-automatic light field editing. We also investigate an alternative navigation interface, which we call *focus*, where the light field is shown with a synthetic shallow depth-of-field. This is based on Davis et al. [27], which relies on defocus, computed following Isaksen et al. [92], to guide the capture of unstructured light fields.

These interfaces rely on different *depth cues* to allow users to visualize the disparity of edits between the views of the light field. While *multiview* uses *parallax* to convey disparity, *focus* relies on *depth-of-field blur* to visualize depth. Both cues are reminiscent of the way people experience 3D in the real world, and it is unclear if one cue is preferable to another, even though most existing light field editing tools have chosen so far to rely on the *multiview* approach. Furthermore, they do not require specialized viewing hardware [242, 150] and do not *intrinsically* necessitate additional data
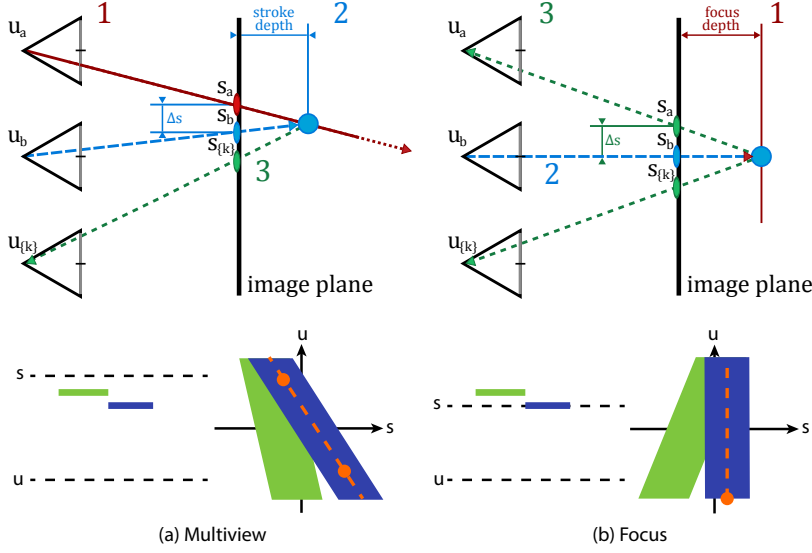
Figure 2.3: Workflow when drawing a stroke in each paradigm. **Top row, left:** *multi-view*. (1) The user first draws a stroke s_a in one view u_a. (2) Depth is then adjusted on a different view u_b by displacing the stroke along the epipolar line. (3) The stroke is then projected onto the other views $u_{\{k\}}$ of the light field, yielding $s_{\{k\}}$. **Top row, right:** *focus*. (1) The user first specifies depth by placing the focal plane. (2) A stroke s_b is drawn on the central view u_b at the specified depth. (3) The stroke is projected onto the other views of the light field, yielding s_a, and s_b. **Bottom row:** A light field interpretation of the two paradigms. In the *multiview* paradigm, the user specifies two correspondences (orange dots), which provide the disparity of the 3D point (slant in the light field). In the *focus* paradigm, the user first places the point of interest in focus by shearing the light field. As a result, all images of the point are aligned and one scribble is enough to edit all the views.

such as scene depth. We also test whether the use of depth information is helpful during editing.

IMPLEMENTATION   The interfaces tested share the same screen layout, shown in Figure 2.2. On the left, there is a description and example image of the current task. Next to it, a control panel and two working windows, named Window 1 (*W1*) and Window 2 (*W2*). Interface manipulation is performed with a mouse or tablet. In the *multiview* interface (Figure 2.2, left), the user is presented with two views of the light field, whose viewpoints are independently manipulated by panning and tilting. This allows the artist to view the light field from a different viewpoint than the one used for manipulation, which is a common workflow in 3D software packages. In this interface, the windows are equivalent. The workflow for placing a stroke in *multiview* is described in Figure 2.3, top left. In the *focus* interface (Figure 2.2, right), the scene is rendered with a wide synthetic aperture that blends all views of the light field [92]. Points that are in focus appear sharp because their images are aligned, while points that are out of focus appear blurry because of the disparity between their images. By construction, this alignment gives us the position of any in-focus point in all views of the light field. In this interface, the user cannot alter the viewpoint, but can adjust the depth of the focal plane of the scene, i.e. the relative disparity of the views [216]. Additionally, with depth enabled, the views focus directly on the visible

surface below the mouse. Here, W2 displays the edit without blending it with the views of the light field, to show its area of influence in other views and help determine occlusions. The workflow for placing a stroke in *focus* is described in Figure 2.3, top right.

Many editing tasks require users to pick locations in 3D space to, e.g., draw strokes. With depth information disabled, we place the user strokes on a plane *parallel* to the camera, so all points in the stroke share the same depth. Otherwise, we snap the strokes at the depth under the pointer. We found no effect due to depth quantization when using depth to edit. In both cases the strokes have zero-width in the depth domain. We refer the reader to the supplementary video² for a practical demonstration of the workflow with each of them.

EXPERIMENTS   We perform two separate user studies. In our first study, we analyze the performance of the two paradigms described above (multi-view and focus) when used exclusively, and considering their use with and without depth information. This yields a total of *four* interfaces. We evaluate those interfaces by making subjects perform several simple tasks. This first study allows us to gain an initial but formal understanding of the light field editing process by precisely characterizing the strengths and weaknesses of each paradigm in simple, controlled tasks.

In our second study, we analyze users' workflow on more complex, but natural, editing tasks, informed by the analysis of the first study. In this case, we let users switch paradigms and activate or not the use of depth; we also include additional tools based on previous feedback. Furthermore, we use captured light fields exhibiting imprecise reconstructed depth.

2.4   EXPERIMENT 1: SYNTHETIC SCENARIOS

GOAL   We want to compare the two interfaces with respect to their effectiveness, efficiency and subjective preference. With *effectiveness* we refer to how well the intended task is accomplished, *efficiency* is related to the effort of obtaining a particular output, and *subjective preference* is based on qualitative data, i.e. user opinions on ease of use, learning curve, among others.

LIGHT FIELDS   We use three different synthetic light fields, depicting different types of scenes (see Figure 2.4): a complex architectural scene (*San Miguel*), a still-life scene (*vase*), and a human head (*head*). These scenes have different depth, geometry and reflectance complexities. We use synthetic scenes to have precise depth information and be able to compute the error of the performed edits. We render the scenes with a light field camera implementation in the physically-based renderer PBRT [176]. We use 17×17 views with a resolution of 400×400, in order to achieve real-time interactions at roughly 30 frames-per-second. We up-sample the rendered images to 600×600 during display to facilitate more accurate placement of the strokes.

EDITING TOOLS   The set of tools for this study consists of: *brush painting*, *erasing*, *dodging & burning* and *pasting* of pre-loaded billboards parallel to the camera plane. All these edits are directly controlled by strokes locations. We choose these operations since they are common in most image editing software, they are well-known to users, requiring little training, and

---

2 http://giga.cps.unizar.es/~ajarabo/pubs/lfeiSIG14/videos/Jarabo2014_main_video.mp4

they represent simple operations from which more complex edits can be performed.

TASKS   We asked users to perform two types of tasks: *directed tasks*, where the user has specific instructions on what to edit; and *open tasks*, where the user is only given general guidelines. We refer the reader to Section 2.A for the specific instructions and example target images.
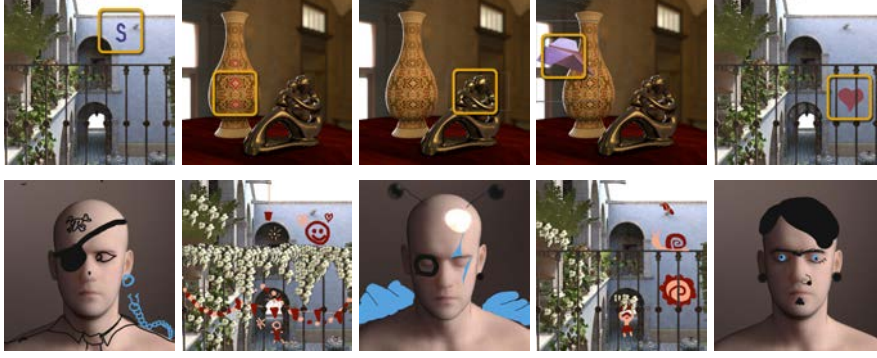
Figure 2.4: Top row: Target images given to users in the *directed* tasks of the first study (S1 to S5). The small highlighted areas have been added to this figure for visualization purposes and future reference. *Bottom row:* Example results of user edits for the *open* tasks, S6 and S7.

Directed tasks are performed for all four interface configurations. We use the central view of an edited light field as the target image. Users are not required to match the target image precisely, but rather to match the depth at which the strokes have to be positioned using the target image as a visual reference. Only one tool is available for each task, plus the eraser which is always available; the color brush is limited to one color, to avoid unnecessary distractions. Time to completion is limited to 5 minutes. These tasks, and the specific light field for each, are the following (see Figure 2.4):

S1 Draw a simple object on a surface of constant depth (*San Miguel*)

S2 Paint a simple pattern on a non-planar surface (*vase*)

S3 Increase the brightness of the specular reflections on a curved surface (*vase*)

S4 Place an object billboard within a certain depth range in free space (*vase*)

S5 Draw on a partially occluded surface (*San Miguel*)

The five tasks have been chosen to cover a wide range of use cases: Tasks S1 and S2 are devised to test general editing of surfaces, while Task S3 deals with the particular case of specular highlights, which do not lie on the surface of the object. Task S4 investigates how to work in free space, while Task S5 tests how to best deal with occlusions.

After performing the directed tasks, subjects complete two *open tasks*, where real-world photos (see Table 2.2) are given as a source of inspiration, and participants are free to use all the tools at will, plus two different colors for the brush. Time to completion is limited to 12 minutes. The tasks vary based on interface selection by the user:

S6   The user is allowed to select whether to use depth information or not during editing. The task is done twice per subject, once with the *multiview* paradigm (toggling freely between using or not depth), and once with *focus* (also with or without depth). The task is carried out on the *head* light field.

S7   The user is allowed to freely change between the four interfaces. The task is done on the *San Miguel* light field.

EXPERIMENTAL PROCEDURE   Twenty paid subjects participated in the experiment (6 female, 14 male). All of them had previous knowledge on image editing, 3D modeling or 3D sculpting software, with either an artistic or technical background. Most of them (90%) had no previous knowledge of light fields, which were briefly introduced to them in the beginning.

Each subject used each interface, with and without depth, first for all directed tasks and then the open ones. We randomize the order of each interface configuration to mitigate learning effects and record all users actions and screen. Subjective preferences were collected with questionnaires filled after each interface and at the end of the experiment. The full experiment took around four hours per subject, including training and breaks.

2.4.1   *Analysis*

Throughout the experiment we collected both quantitative data on task errors and timings, qualitative data on performance and difficulty of both tasks and interfaces, and free-form comments on interface effectiveness. We report here the analysis and main findings and include additional data in Section 2.B. For brevity, we will refer to our interface configurations in the rest of the analysis as *M* (multiview without depth), *MD* (multiview with depth), *F* (focus without depth) and *FD* (focus with depth).

PROCEDURE   We use repeated measures ANOVA for the analysis of error, timings and ratings, to test whether the levels of a factor (e.g. interface is a factor; the types of interface are its levels) have influence on the observed data. It is a repeated measures scheme because we measure the same independent variables (e.g. error) using the same participants under the different conditions. Additionally, Kruskal-Wallis (a nonparametric extension of ANOVA) is used to analyze rankings, since they are an ordinal variable and typically cannot be assumed to follow a normal distribution. In all tests, we use a *p*-value of 0.05 to indicate significance. When sphericity is violated, according to Mauchly's test, we report significance values adjusted with the Greenhouse-Geisser correction [24]. In all figures, error bars represent the standard error of the mean. We perform outlier rejection on the measured error data, based on the interquartile difference, with a factor of 2.2 [81]. This led to dropping one user in Tasks S2, S3 and S5, two users in Task S1 and three users in Task S4.

ERROR IN DEPTH   To evaluate how well a user can specify locations in the light field, we measure the *error in depth* of the stroke. We choose this over measuring image-based differences since our tasks are not pure matching tasks. Specifically, for each view of the light field, we first compute the L_1 distance between the depth of the stroke and the target depth, for each pixel of the stroke, and divide it by the number of pixels covered by such stroke. We then average across all views of the light field. Our experiments showed

that L_1 averaged across views approximates a normal distribution better than other metrics, which facilitates the subsequent analysis. Note that in Task S4 (positioning in free space) there is not a single fixed target depth, but a valid range between the vase and the sculpture. We thus compute error in depth with respect to the limits of such range, assigning a value of zero within it.

Figure 2.5 (top) shows the per interface mean error for each of the directed tasks (S1–S5). The error is highly dependent on the task, which accounts for 73% of the variance. When taking into account interfaces, the ANOVA yielded significant differences between interfaces for all tasks, as summarized in Table 2.1. The figure additionally illustrates significant differences between interfaces according to the pairwise comparisons tests.

This analysis suggests that while interfaces with depth information work well when manipulating surfaces without occlusions, not using depth information is actually more effective when occlusions are present, or when the editing task requires positioning in free space. This is due to the fact that the edits will snap to the underlying depth of the light field, which is not desirable in those particular cases. A deeper analysis on workflows can be found in Section 2.6. Moreover, the supplementary material contains a video[3] showing Task S5 (handling occlusions) being performed with all four interfaces, as well as Tasks S1 to S4 performed with different interfaces.
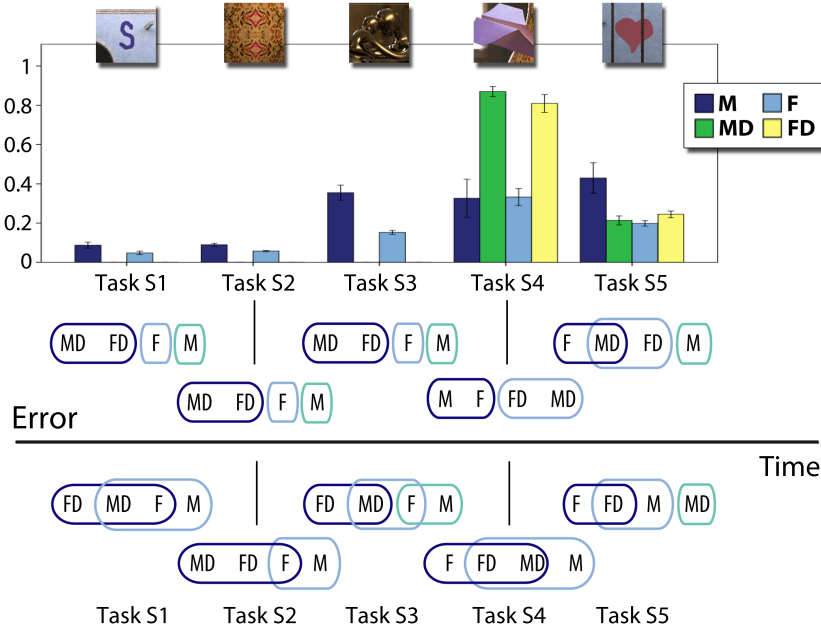
Figure 2.5: Top: Mean error per interface for each task and pairwise comparisons for the error in each task. Items in the same set are statistically indistinguishable. Bottom: Pairwise comparisons for the time to completion in each task.

TIME TO COMPLETION   We illustrate statistically significant differences between interfaces in time to completion for each directed task (S1–S5) in Figure 2.5 (bottom). In general, users were able to complete the tasks in the allocated time with all interfaces. We observe that having depth information

---

3 http://giga.cps.unizar.es/~ajarabo/pubs/lfeiSIG14/videos/Jarabo2014_suppvideo_Experiment1.mp4

	H	(df_1, df_2)	p	$\eta^2(\%)$
S1	25.230	(1.517, 25.792)	0.000	59.7
S2	138.745	(1.305, 23.491)	0.000	88.5
S3	70.390	(1.612, 20.108)	0.000	79.6
S4	24.951	(1.861, 29.779)	0.000	60.9
S5	6.275	(1.264, 22.760)	0.015	25.8

Table 2.1: Results of the repeated measures ANOVA for the interface factor for the error in depth in each of the tasks. H is the test statistic, df_1 and df_2 the between-group and within-group degrees of freedom, respectively, p the associated significance and η^2 is indicative of the proportion of variance of the data that the interface factor explains.

leads to faster editing when painting on surfaces, as long as no occlusions are present; *M* tends to take longer than the rest, apparently being less intuitive for users, while *F* performs well in most situations, specially when dealing with occlusions or positioning in free space, which are the two most challenging scenarios in our tests.

RANKINGS AND RATINGS   The final questionnaire contained eleven questions in which the users had to rank and rate the four interfaces. Five questions referred to the preference of interface for each of the five directed tasks, and one to the overall preference. The remaining five questions investigate preference in more general aspects, namely positioning in depth and on a plane (x-y), erasing, perceived accuracy of the interface and difficulty of use. For each ranking, we also compute the rank product per interface $\Psi(\vartheta) = (\prod_i r_{\vartheta,i})^{1/m}$, where $r_{\vartheta,i}$ is the ranking received by interface ϑ in a specific question and m the number of subjects [191]. We use rank products to sort the interfaces when grouping them in statistically different groups (Figure 2.6, top).

Rankings for the different tasks (Figure 2.6, top) exhibit again a large between-task variability, in accordance with the error and time to completion. In Tasks S2 and S3 *MD* and *FD* are ranked significantly higher ($p \leq 0.035$) than no-depth interfaces (*M* and *F*). The difference between the interfaces with depth (*MD* and *FD*) is not significant ($p = 0.160$). In Tasks S4 and S5 the trend is again reversed: there is a clear preference for interfaces without depth, and in particular for *F* ($p \leq 0.035$ and $p \leq 0.008$ in Tasks S4 and S5, respectively).

When asked about the overall ranking, *MD* ranks first, significantly higher than *FD* and *M* ($p \leq 0.011$), but there is no significant difference between the rest. This is probably due to the large dependency on the task, shown by previous analyses. Despite the similarity of both interfaces in the rest of the tests, the users reported that the *multiview* paradigm allowed them a better visualization of the light field and the edits. Results of pairwise comparisons between interfaces for mean ratings for preference per task and overall preference are shown in Figure 2.6 (bottom). These ratings strongly correlate with rankings (Spearman's $\rho = 0.80$, $p \simeq 0.000$). This is meaningful, indicating that users have clear opinions regarding the interfaces for the different tasks.
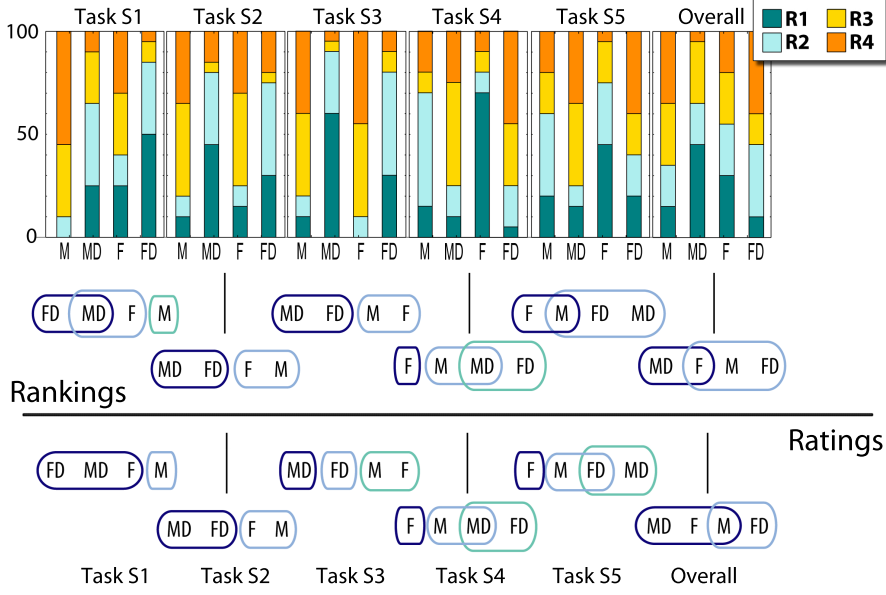
Figure 2.6: Top: Rankings from final questionnaire for questions on preference for each task and overall preference (R*i*: rank *i*), and pairwise comparisons between interfaces for the rankings. Items in the same set are statistically indistinguishable. Bottom: Pairwise comparisons between interfaces for the ratings. There is a clear agreement between both ratings and rankings.

PREFERENCES IN OPEN TASKS   During open tasks, when users can freely toggle the use of depth and interface, we record the time spent on each interface, and what actions are performed in each of them. Specifically, we track the time they spend drawing, erasing, changing the view point, and adjusting depth. When given complete freedom, the preferred workflow is to perform edits with the *focus* paradigm, use *multiview* to inspect the changes from different viewpoints, and then go back to editing with *focus* again. This is supported by Table 2.12, which shows the total number of times users went from one interface to another during editing, and by the users' feedback in the final interview, where they agree in their preference to edit with *focus* due to the feeling of control it provides. Finally, based on their answers to the after-task questionnaires, 85% of the users found the possibility of switching between interfaces during Task S7 helpful or very helpful (as indicated by a rating of 4 and 5 on a scale from 1 to 5). We refer to the supplementary video[4] for sample editing sessions by subjects for these tasks.

MAIN FINDINGS AND INTERFACE IMPROVEMENTS   In this experiment we make the following observations, which lead to improvements in our editing system for the next experiment:

(i) There is a high between-task variability with respect to the choice of interface in user preferences and error. Furthermore, the vast majority (85%) of users report that switching between paradigms and toggling depth on/off is very helpful.

$\rightarrow$ We merge both interfaces into one in which users can switch between focus and multiview, and turn depth on or off.

---

4 http://giga.cps.unizar.es/~ajarabo/pubs/lfeiSIG14/videos/Jarabo2014_suppvideo_Experiment1.mp4

(ii) Handling occlusions is challenging for users, to the point that they prefer not using depth information if they are present.
→ We introduce a *Depth Selection* tool that creates a selection mask by clicking on a location to set the depth and specifying a depth threshold; this avoids unwillingly editing occluding parts when trying to edit occluded ones.

(iii) Focus gives users better understanding of where they are positioned in the light field.
→ We introduce a *Visual Aid* mode that helps visualizing the active depth range (either the one selected or the one potentially selectable, if a threshold is set) by placing a semi-transparent checkerboard on the parts of the light field which are not within the active depth range.

(iv) In interviews, users demanded selection tools based on color to be combined with depth-based ones.
→ We introduce a *Color Selection* tool that creates a selection mask based on picking a color and setting a threshold around it in color space. This threshold is independent of the depth threshold, and thus each one yields a different selected area. This tool can be combined with the *Depth Selection* tool, in which case the selected area is the intersection of both.

2.5 EXPERIMENT 2: REAL SCENARIOS

GOAL   In the second experiment, we investigate common edits on real, captured light fields with *reconstructed* depth. Our goal is to evaluate whether users can perform such edits in the presence of imprecise depth, and what interaction paradigms and tools they prefer. We incorporate the changes mentioned above and also include a *Hue Brush* for more versatility. Quantitative evaluation of the effect of imprecise depth could in principle be performed by introducing errors in depth procedurally; while this would allow a controlled error analysis, we found that the different existing depth-reconstruction algorithms produce very different types of artifacts, which cannot be modeled with simple functions. Thus, such quantitative analysis would hardly generalize to real scenarios. We choose instead to test whether such reconstructed depth maps from a varied set of real scenes are good enough to allow users to satisfactorily perform convincing edits.

LIGHT FIELDS   We use eight captured light fields, depicting different types of scenes, shown in Figure 2.7. The first scene, *watch*, is captured with a Raytrix camera and its depth obtained with the algorithm by Wanner and Goldlücke [230]. The next four—*couch*, *church*, *statue* and *mansion*—are captured with a camera gantry, and depth information is obtained with the algorithm by Kim and colleagues [117] (we use light fields and depth maps from their own database). The final three scenes—*matrioska*, *lab*, and *frog*—are captured with a Lytro light field camera, and the depth maps obtained with the Lytro SDK. These scenes have different depth, geometry and reflectance complexities, feature different baselines, from small (Lytro and Raytrix) to large parallax (Kim et al.) between views, and cover a wide range of capture setups and depth reconstruction methods found today. Light field angular resolutions are 9×9 in all cases except for the horizontal-parallax-only ones [117], whose resolution is 17×1. Spatial resolution is the same as in the first study.
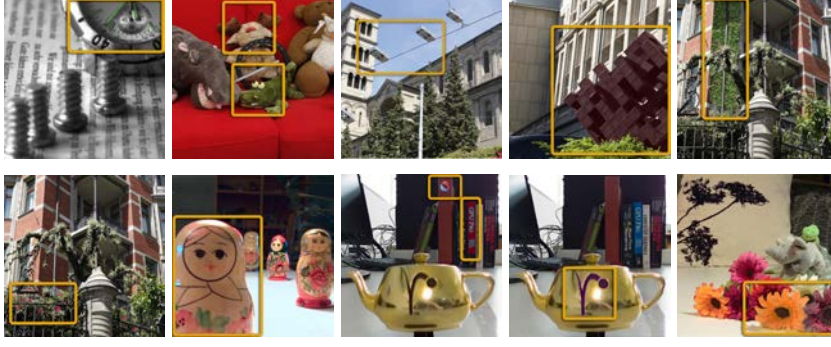
Figure 2.7: Target images given to users in the tasks of our second study. The small highlighted areas have been added to this figure for visualization purposes, marking the area to edit. In reading order, Tasks R1 to R10. Different depth reconstruction methods are used: R1 is based on Wanner and Goldlücke [230], R2 to R6 are based on Kim et al.[117], and R7 to R10 are from Lytro.

TASKS   We asked users to perform a variety of *directed tasks* (see Figure 2.7) where textual instructions and an example target image (corresponding to the central view of the light field) are given to the user, who is asked to perform the given edit as accurately as possible. We chose tasks that span a wide range of use cases, in terms of editing operations, tools used, complexity of the geometry of the light field, quality of the depth reconstruction, and type of edition (occlusion, free space, curved surface, etc):

R1 Colorization: Colorize the arrows and the marks on the inner circumference of the watch, so that users draw on a slanted surface (*watch*)

R2 Correcting small parts: Change the color of the reindeer's nose and of the eyes of the crocodile, so that users draw on curved surfaces (*couch*)

R3 Cloning an object: Place another light, so that users work in free space (*church*)

R4 Altering a material or color: Change the hue of the statue, so that users deal with selection and complex geometries (*statue*)

R5 Texture brush: Add ivy to the wall, so that users deal with occlusions and slanted surfaces (*mansion*)

R6 Importing an object (billboard-like): Add flowers to the bush, so that users deal with occlusions and work on areas of complex reconstruction (*mansion*)

R7 Change luminances in the scene: Use dodge to brighten-up the matrioska in the foreground, so that users edit curved surfaces with a coarse depth reconstruction (*matrioska*)

R8 Altering the color of an object: Change one of the books' color from blue to red, to deal with color selection. Users are also asked to paste a SIGGRAPH logo on a book, another import-and-paste task (*lab*)

R9 Improving small details: Change the RenderMan logo in the teapot to a purplish color to make it more salient (*lab*)

R10 Harmonizing the colors in the scene: Change the colors of the two foremost flowers in the scene to match those of the rest of the flowers, so that users deal with selection of complex geometries (*frog*)

EXPERIMENTAL PROCEDURE   Ten subjects participated in the experiment (four female, six male). All of them had previous knowledge on image editing, 3D modeling or 3D sculpting software, with either an artistic or technical background. Half of them had previous knowledge of light fields. The full experiment took around two hours per subject, including training and short breaks. The order of the tasks is randomized to compensate for possible learning effects. We recorded the screen during all the experiments. Each task had a time to completion limited to ten minutes. After finishing each task, users had to fill in a five-question questionnaire and at the end of the ten tasks the users completed a final questionnaire. All questionnaires can be found in the supplementary material at the project page[5]. Each participant completed the study with a debriefing.

DATA   In this case, since we have no ground truth depth, we cannot measure error. Our objective measurements correspond to time of use of the different interaction paradigms (focus/multiview, with/without depth), the different tools, and the time to completion. Subjective ratings were collected in the questionnaires. For each task, we asked to rate, on a scale [1..5], the difficulty of the task, the similarity of the obtained result to the target, the ability to correctly position editions in space and depth, the perceived inaccuracies in the depth information, and whether those inaccuracies affected the editing process. Finally, we asked them to rate and rank each task according to their difficulty. In addition, we record all their actions.
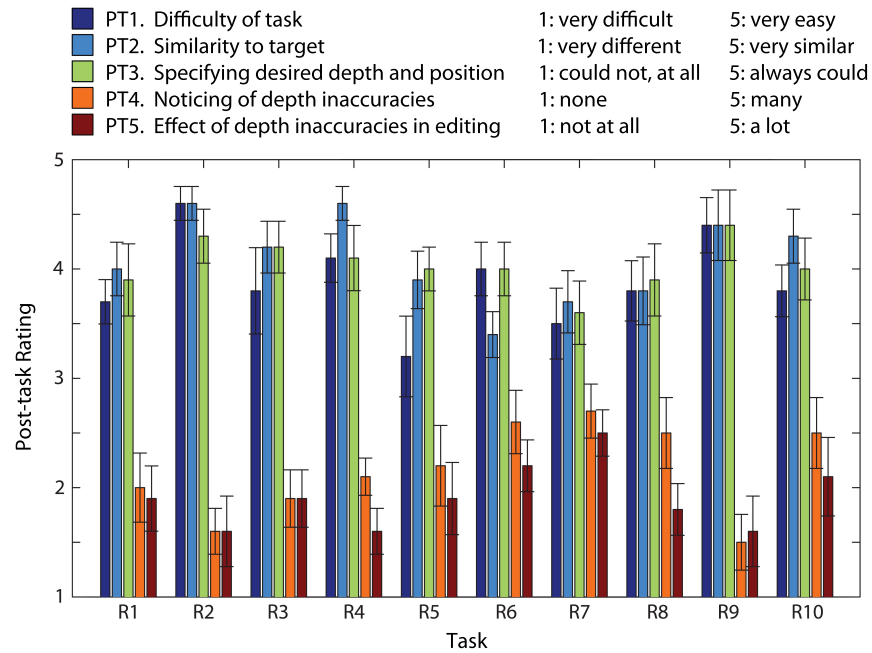
Figure 2.8: Ratings in post-task questionnaires in our second study, averaged across subjects, for each of the five questions. Bars represent the standard error of the mean. Note that in questions one to three higher is better, while in the two last questions lower is better.

2.5.1 *Analysis*

OVERALL   In general, subjects were able to perform the tasks satisfactorily in the allotted time, as indicated by results for post-task questions (see Figure 2.8 for the results and questions): All subjects were satisfied with the result obtained (mean rating across subjects and tasks for similarity to the target is $\mu_{PT2} = 4.09$, 5 being "very similar" and 1 "very different"), and they did not find the editing very difficult (mean rating across tasks and subjects for difficulty is $\mu_{PT1} = 3.89$, 5 being "very easy" and 1 being "very difficult"). Further, they state that they were "almost always", $\mu_{PT3} = 4.04$, capable of placing the edits in the (3D) position they intended.

INTERACTION PARADIGMS   Figure 2.10 (left) shows the mean times, across subjects and per task, during which the focus and multiview paradigms were used, in percentage with respect to time to completion of the task. Differences are not significant for any task except for task R3 (*church*), where multiview is preferred, and task R7 (*matrioska*), in which focus is preferred. The first one is to be expected, since blur (or de-focus) does not offer a clear depth cue due to the nature and configuration of the scene. In R7, however, focus is more frequently used, the difference with respect to multiview being significant; we suspect this is due to the scene having a small number of very clearly separated depth planes.

In the final questionnaire, subjects rated and ranked multiview and focus, both in terms of visualization and of editing per se. For visualization, multiview is required by almost all subjects (see mean rating in Figure 2.9, Q2), in a manner consistent with results in our first study; also consistently, rank product for multiview for visualization is 1.07, meaning most of the time it is ranked first of the two. However, when it comes to editing, subjects are almost divided between both paradigms, meaning there is no preference for one over the other (rank product for multiview for editing is 1.62). When asked in the debriefing, many subjects would state that, given the tools at hand and the Visual Aid, there was not a significant difference between one paradigm and the other for editing.

DEPTH INFORMATION   With respect to the use of depth information we consider two aspects: First, whether inaccurate depth is good enough for editing light fields; and second, whether the use of depth information is preferred. When looking at post-task questions, perhaps surprisingly, for most of the tasks users noticed almost no inaccuracies in the depth information (mean rating is $\mu_{PT4} = 2.16$, with 5 being "many inaccuracies" and 1 being "none"). More importantly, the inaccuracies noticed did not significantly affect their editing (mean rating for the corresponding question is $\mu_{PT5} = 1.91$, 5 being "they affected editing a lot" and 1 being "they did not affect editing at all"). These averages are slightly lower for the more sophisticated method of Kim et al. [117], which also makes use of higher resolution light fields, than for the Lytro light fields ($\mu_{PT4,Kim} = 2.08$ vs. $\mu_{PT4,Lytro} = 2.30$, and $\mu_{PT5,Kim} = 1.84$ vs. $\mu_{PT5,Lytro} = 2.00$), although the difference is not significant. These conclusions on depth inaccuracies are confirmed by subjects' responses in the final questionnaire (see Figure 2.9, Q5 and Q6). In terms of usefulness of depth information, users' opinion has improved in the second study. When looking at the final questionnaire responses (Figure 2.9, Q3 and Q4), we see that they find it "useful" or "very useful" and use it "frequently" or "very frequently" (more detail in Section 2.6). Timings confirm

this, as shown in Figure 2.10 (right), which depicts the mean percentage over total time, across subjects and per task, during which depth information is activated or de-activated. This confirms the need for taking into account depth information but coupled with tools that exploit it adequately. It should be noted that while the task performed and the reconstruction quality might influence users preferences, it is not our goal to perform an accurate evaluation of depth reconstruction methods, but to derive usable guidelines for light field editing taking into account state-of-the-art capture and reconstruction methods.
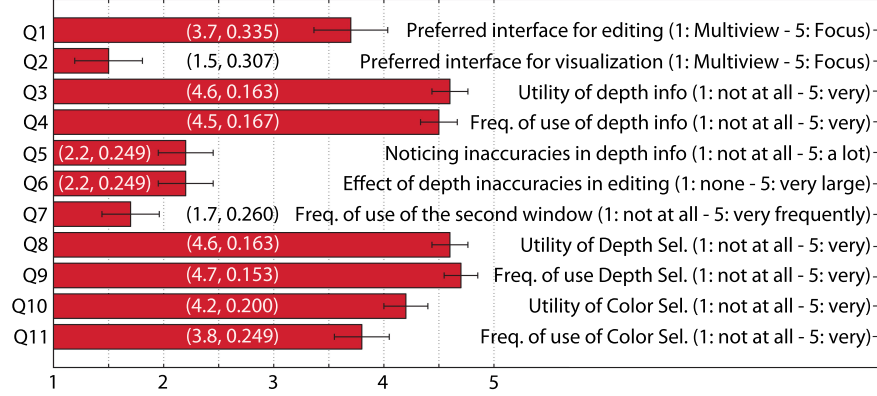
Figure 2.9: Mean ratings for questions in the final questionnaire of our second experiment, averaged across subjects. Bars represent the standard error of the mean (SEM). For each question, the exact mean μ and SEM are shown in the form (μ, SEM). Note that not always higher is better.

NEWLY INCORPORATED TOOLS   When asked about the Depth Selection tool in the final questionnaire, responses by subjects were almost unanimous: the vast majority find it "useful" or "very useful", and use it "frequently" or "very frequently" (see Figure 2.9, Q8 and Q9). Similar to the Depth Selection tool, the Color Selection tool is well received by subjects (Figure 2.9, Q10 and Q11; mean rating for utility of Color Selection is 4.20, 5 being "very useful" and 1 being "not useful at all"), although they state that the use of the Color Selection tool is less frequent than the Depth Selection tool ($\mu_{FQ,DepthUse} = 4.70$ vs. $\mu_{FQ,ColorUse} = 3.80$). The tasks are designed in such a way that, four of them can be done only with Color Selection (R2, R8, R9, R10), four of them require Depth Selection (R1, R5, R6, R7), and one both (R3 does not require either of them). Yet, the frequency of use of the Depth Selection tool is significantly higher, as stated above, meaning that subjects tend to favor the Depth Selection, or a combination of both even if only Color Selection could suffice.

We also look at the temporal data to evaluate the use of the newly incorporated features: Depth Selection, Color Selection, and Visual Aid functionalities. The times of use show that in most of the tasks, at least 50% of the time either the Depth Selection or the Color Selection tools (or both) were activated, as shown in Figure 2.11 (left). The frequency of use of the Visual Aid feature can be seen in Figure 2.11 (right); on average, the time Visual Aid is activated exceeds 50% of the time depth information is used in all tasks, confirming its utility. Section 2.6.2 analyzes these workflows in more detail.

CONCLUSIONS OF EXPERIMENT 2   We have seen that subjects are able to satisfactorily perform the variety of tasks with the interface and tools provided in the allotted time. Surprisingly, they very seldom notice inaccuracies in the depth information, and in any case these do not significantly affect their editing process. Further, for editing, there is not a significant difference between focus or multiview, whereas as expected, for visualization an approach that lets users see the different views individually is favored. We can conclude that for editing, provided the adequate tools and depth management functionalities (e.g. Visual Aid and Depth Selection), the differences between the two interaction paradigms are blurred out. Another relevant finding is the fact that users do leverage the extra dimensionality of a light field: Even in cases where selection based on color would suffice, they complement it with selection based on depth. Consequently, depth information, together with depth management functionalities, are favored and extensively used, easing previously complicated tasks such as occlusion handling.
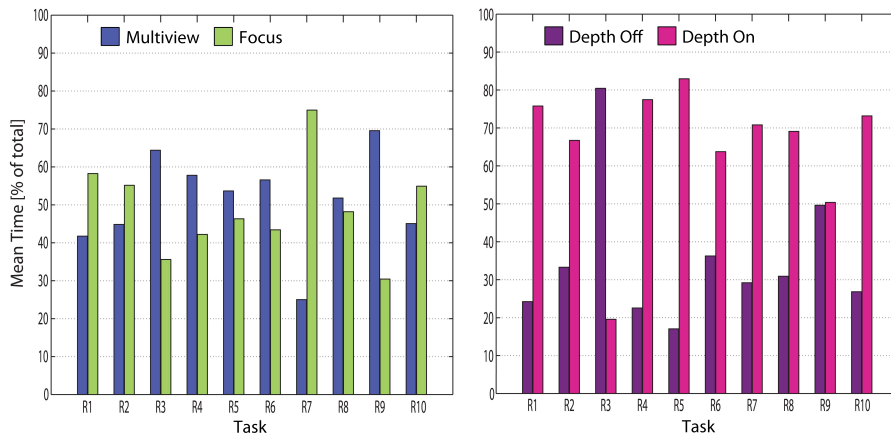
Figure 2.10: Mean percentage across subjects, in realistic scenarios, of total time spent with multiview or focus (left) and with depth activated or deactivated (right), with respect to the total time to completion.
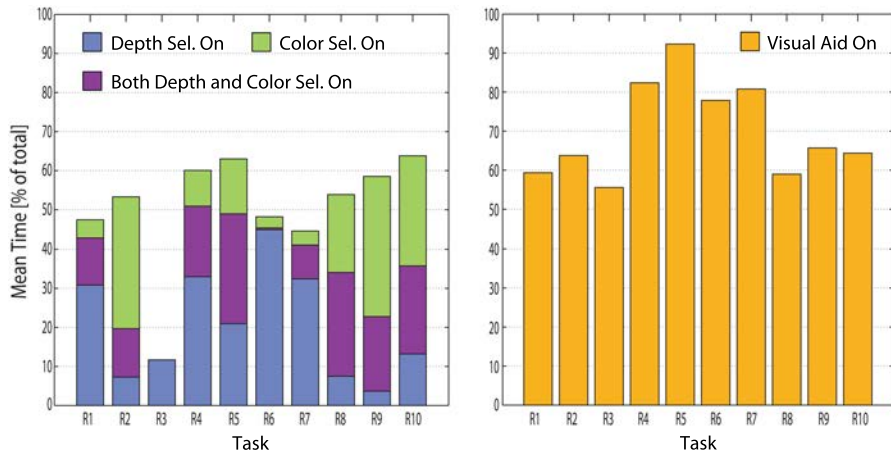
Figure 2.11: Left: Mean across subjects of time of activation of only Depth Selection, only Color Selection, and both tools simultaneously, in percentage over total time to completion of the task. Right: Mean times of use of the Visual Aid, in percentage over total time using depth information.

2.6 WORKFLOW ANALYSIS

Data collection from the experiments described above yields an immense quantity of both subjective and objective information which provides insights on a variety of different aspects of light field editing: Usage of different tools, preferred interfaces, variability of preferences with the task to perform, workflows, etc. While previous sections have focused on how suitable different interfaces and tools are, and whether an interface can allow for satisfactory editing of light fields with current depth reconstruction methods, here we analyze subjects' workflows, that is, we look for underlying patterns in subjects' actions and their preferences for different generic tasks.

2.6.1 *Synthetic Scenarios (Experiment 1)*

Analysis of the data for time to completion, error in depth (measured as the L1 norm averaged across views), and ratings and rankings on interface preference provided by users yielded three distinct clusters, roughly corresponding to three task categories: editing of surfaces (planar or curved), editing in free space, and occlusion handling. Data analysis is performed using repeated measures ANOVA for error, timings, and ratings, and Kruskal-Wallis for rankings (see Section 2.B.1).

EDITING OF SURFACES   Tasks S1 to S3 allow us to draw insights on surface editing. In terms of error, in these tasks error in depth for interfaces with depth (*MD* and *FD*) is zero, since strokes snap to the surface below them (Figure 2.5, top). In a consequent manner, realizing the task with these interfaces took less time (Figure 2.5, bottom). For interfaces without depth, *M* yielded a higher error ($p \leq 0.018$) than *F* ($p \leq 0.018$), showing that users found it more difficult to locate an edit in depth with *M*. This is also reflected in the timings, in which *M* is significantly slower ($p \leq 0.008$) in most cases. Task S1 is an exception, possibly because of its simplicity. Ratings show a clear preference for interfaces with depth for editing on surfaces, as expected. The correlation of rankings with the results for ratings is extremely high, showing that subjects have clear preferences. In summary, interfaces with depth are the clear choice for editing surfaces, as expected. Further, a slight preference for multiview over focus is hinted, but the difference is not significant

EDITING IN FREE SPACE   Task S4 deals with positioning in free space. In this case, interfaces without depth yield lower error than those with depth, although the difference between interaction paradigms is not significant (Figure 2.5, top). Even though errors are high, times to completion is very low (Figure 2.5, bottom). In interfaces with depth, this can be due to the fact that users realize that they will not be able to correctly place it in depth (recall that the edit will snap to the surface right below) and give up. This hypothesis seems supported by the low ratings these two interfaces receive in this task. In general, results suggest that users struggle to correctly judge depth in free space, and feedback from users confirmed this fact, and pointed out that *F* is chosen because of the clear feedback it provides with respect to the position of the plane being edited. In accordance with this, *F* takes the least time, even though the difference is only significant with respect to *M* ($p \leq 0.008$). Thus, *F* is the interface of choice among the four tested for editing in free space.

OCCLUSION HANDLING   Task S5 implies handling occlusions. For this, *F* yields both the lowest error and time to completion (although not significantly different from *FD* in the latter and from *MD* in the former) (Figure 2.5). Clearly, *M* and *MD* are not useful interfaces in this regard, since *M* yields the largest error, and *MD* the highest time to completion, to the point that some subjects (35%) did not complete the task in the given time. Note that in this experiment, subjects need to erase to handle occlusions, which can be time consuming. Ratings confirm the superiority of *F* in this task.

In summary, this experiment has shown the large influence of the task to perform in the interface of choice, with, essentially, *MD* and *FD* being ahead for on-surface editing, and *F* the interface of choice for free-space editing and occlusion handling. In the following, we will see if these findings hold in real light fields, with new tools and subjects being freely allowed to choose the interaction paradigm and whether depth information should be used (essentially, being able to switch between the four previous interfaces).

2.6.2 *Real Scenarios (Experiment 2)*

Again, we cluster the tasks in categories; these are the same as the above, with an additional category for editing of objects of intricate geometry, for which subjects follow a different workflow than when editing simpler planar or curved surfaces. As mentioned above, our focus here will be on workflows, and we will look at times of use of the different interfaces and tools as an indicator of preferences. We also collect data on subjective preferences. We describe here the main findings, while the rest of the data can be found in Section 2.C. Additionally, we refer to the supplementary video[6] for examples on the workflows.

EDITING OF SURFACES   Tasks R1 and R8 involved editing planar surfaces, while R2, R7 and R9 require editing curved surfaces (see Figure 2.7 for reference). Among the latter, R2 and R9 are very similar in nature, since they both involve changing small details, while R7 requires editing a large curved surface. In all of these tasks, the use of depth information is largely favored, which matches our results in Experiment 1. There is no clear preference between interaction paradigms (focus or multiview), although there is a slight trend towards focus; in the debriefing interviews, subjects reported that focus offered a very strong and easy-to-interpret cue for visualization of the active area. Regarding the tools used, Tasks R2 and R9 favor the use of the color selection tool, possibly because, as mentioned, the areas requiring editing are small, similar in color, and without a distinct depth with respect to their surrounding areas. The rest favor the use of the depth selection tool. A sample editing workflow (for Task R1) is shown in Figure 2.12.

EDITING IN FREE SPACE   Task R3 requires editing in free space. In this case, again, results are consistent with Experiment 1: Depth information is scarcely used, if at all. However, while, before, *F* was preferred, we observe here a trend towards *M*, as shown in the workflow for Task R3 included in Figure 2.12. This is possibly due to the absence of high frequency information in the area to edit, which causes the blur of the focus interface to provide little or no depth cues.

---

6 http://giga.cps.unizar.es/~ajarabo/pubs/lfeiSIG14/videos/Jarabo2014_suppvideo_Experiment2.mp4
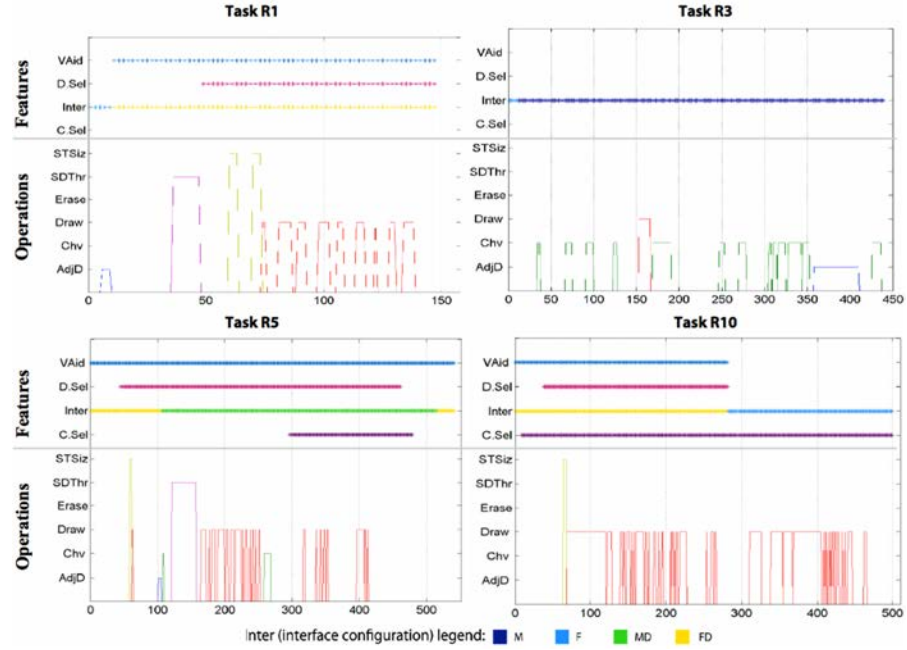
Figure 2.12: Sample workflows for Tasks R1, R3, R5 and R10, one pertaining to each of the four categories or use case scenarios (editing surfaces, editing in free space, handling occlusions, and editing complex geometries). They indicate the tools and interfaces used by the subject along time, shown in the abscissa (in seconds). The six bottom marks in the y-axis correspond to different operations (from bottom to top: adjust depth, change view, draw, erase, set depth threshold, and set tool size). The top four marks indicate whether that feature was activated or not at each time instant, and correspond, from bottom to top, to color selection tool (*C.Sel*), interface (*Inter*), depth selection tool (*D.Sel*) and visual aid tool (*VAid*). Note that the Inter features different colors specifying the interface being used.

OCCLUSION HANDLING   In this experiment, Tasks R5 and R6 require dealing with occlusions. Here, the introduction of the depth and color selection tools causes a change with respect to results obtained in Experiment 1. While in the first experiment, there was a large amount of erasing to deal with the occlusions, the introduction of the depth selection tool, largely used in both R5 and R6, reduces the need to erase to a minimum (see Figure 2.12, Task R5, for a sample editing workflow in that task). Surprisingly, there is little difference between the use of *MD* and *FD*, revealing that as long as depth information and related tools are present, the interaction paradigm is less relevant for these tasks. The color selection tool is fairly used in Task R5 to avoid the pipe, which is hard to disambiguate from the rest of the wall in the depth dimension.

EDITING OF COMPLEX GEOMETRIES   Tasks R4 and R10 clearly show the need for the color and depth selection tools. When intricate geometries are present, these are extensively used. The nature of the scene determines which one is used: In the case of R4, 9 out of 10 subjects used the depth selection tool to complete the task, while in the case of R10, 9 out of 10 used the color selection tool, as shown in the sample editing workflow for Task R10 shown in Figure 2.12. The majority of the subjects used depth informa-

tion throughout the tasks; however, differences between the time of use of *FD* and *MD* were not significant.

In summary, the second experiment confirms the findings of Experiment 1 in most aspects, with a clear exception in occlusion handling, which is now easily dealt with thanks to the new tools. Similarly, handling of intricate geometries is possible thanks to these tools. We also observe that depth information is almost always required, while the differences between the interface paradigms (multiview and focus) become less significant.

2.7 EFFICIENT PROPAGATION OF LIGHT FIELD EDITS

Until now, we have focused on point-based editing, where the user needs to carefully position the edits on the light field. While this type of interaction is the building block for other operations, it limits what the user can do without long and painful editing sessions and strong artistic skills. Here we briefly describe the work we have developed on edit propagation for the multidimensional domain of light fields, and refer to the publications for details [100, 6].

Coarse stroke-based edit propagation [140, 174, 4, 251] allows fast editing in images by asking the user to input just a few sparse strokes, and propagating them to the rest of the image. This propagation is usually performed keeping in mind two principles: first, pixels covered by one stroke should keep the appearance given by the user as much as possible. And second, near pixels with similar appearance should receive similar edits. To account for these two principles, it is necessary to define the mathematical formulation of the propagation, where the final propagated edits in a pixel depends on the explicit edit performed over that pixel, and also in the edits performed over pixels with similar appearance.

Editing the appearance of a light field is challenging because of some inherent difficulties. First, the consistency between views has to be preserved. Additionally, the amount of data to manipulate is usually very large, since the size of the light fields tends to be very high in order to provide well-sampled data. Managing such data sizes can make the editing process very slow, which is unacceptable if we are aiming to provide an interactive editing process to the user. Our light field editing framework succeeds in solving these difficulties: it provides a solution which maintains the coherence between the samples in the light field, by taking advantage of the fact that coherent elements in the light field should receive similar edits, as far as they have similar appearance; and it is done very efficiently, giving user feedback in interactive times, by using a downsampling-upsamping approach where similar and close rays are clustered together, and the propagation is performed in the downsampled version of the light field.

We downsample not only in the spatial domain, but also accounting for other features, ensuring that similar (in a multi-dimensional way) pixels will receive similar edits. This implies downsampling the data considering all the dimensions of the distance metric used when propagating edits [135, 140, 4]. To do this, we follow an approach similar to the work of Xu et al. [251], mapping all pixels to an *n*-dimensional affinity space defined by the distance metric and then performing the downsampling in that space. We explore two methods for downsampling: a hierarchical binary space subdivision [100], and a *k*-means [255] based approach [6]. These two methods exhibit competing advantages: while the former is very fast, the latter results in a more optimal clustering of the rays of the light field, at the cost of

more expensive computations. This downsampling is performed only once as a preprocess.

Then, during the editing, the strokes specifying the edits are transformed into the downsampled space, and propagated in that space using the technique by An and Pellacini [4], that allows global propagation of the edits in the scene by taking into account the low-rank nature of the per-pixels-pair distance matrix. Finally, we upsample the propagated results into the original light field using an upsampling procedure similar to the one proposed by Kopf et al. [125].

Figure 2.13 shows some preliminary results on light field edit propagation using our approach. Although it shows the potential of the approach, significantly more work is still needed to increase the robustness and efficiency of our approach: advanced techniques for large-scale low-rank matrix decomposition, together with specific distance metrics for light fields, are a promising avenue for solving the problem.

(a) Original+Scribbles   (b) Kd-Tree [251]   (c) Sparse Ctrl [252]   (d) Ours
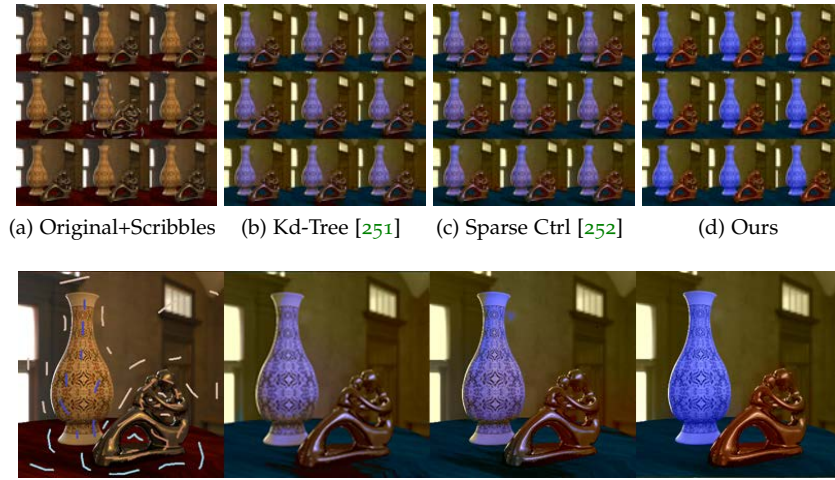
Figure 2.13: Comparison of the results for light field edit propagation using the kd-tree based method by Xu et al. [251], the sparse control method by Xu et al. [252], and our method. The bottom row shows the central view of each light field: Our method propagates more faithfully the input colors from the user and results into proper color segmentation based on the affinity of the different areas of the light field, while the results of the kd-tree [251] and sparse control [252] methods exhibit clear artifacts in form of blended colors, or wrongly propagated areas.

2.8   DISCUSSION AND CONCLUSIONS

Our findings support the potential for light field editing, currently underdeveloped and not on-par with image editing software. Overall, the most important general findings are: First, that users can perform editing tasks on light fields with our interface and tools; and second, that they do exploit the extra dimensionality of the light field. In fact, the real light fields in the second experiment incorporate reconstructed depth information, which subjects used constantly. Surprisingly, even though these reconstructed depth maps have different degrees of inaccuracy, users reported that they barely noticed these inaccuracies, or that they did not significantly affect the editing process. We believe this is due to the nature of the inaccuracies and to
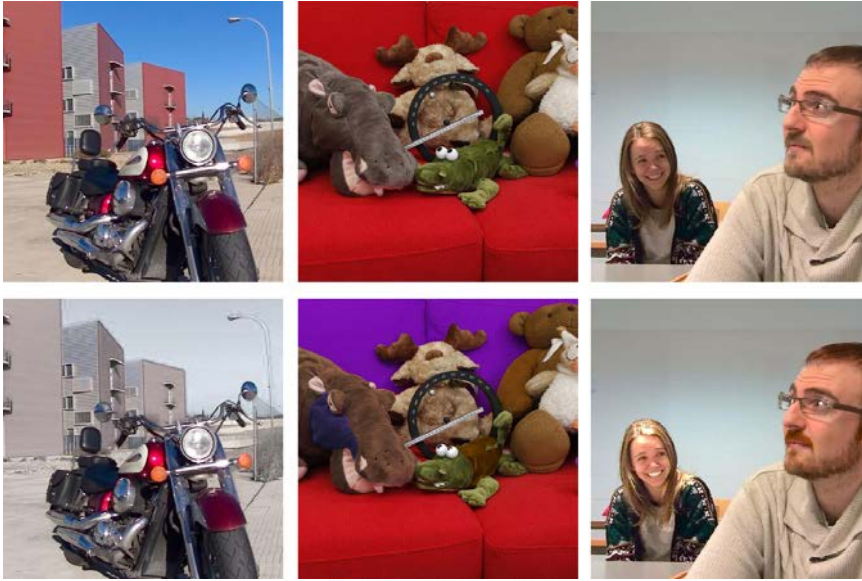
Figure 2.14: Top row: Central view of the original light field. Bottom row: Sample edits performed by advanced users. From left to right: desaturating the background in *motorbike2* (Lytro); modifying the couch, the hippo, and the crocodile's body in *couch* (Kim et al. [117]); and changing his hair color and lighting her up in *couple* (Lytro).

the set of tools provided. For instance, a combination of Depth Selection and Color Selection typically overcomes the influence of noise. This is true even in low-parallax light fields, which do not allow for geometry recovery beyond a series of coarse discrete depth planes.

Moreover, occlusions, which are a very specific and relevant aspect in light field editing, were handled gracefully using the Depth Selection tool in the second experiment. However, having depth information does not mean using it all the time. In fact, users reported that toggling depth off offered them a *high degree of control* over the edits, although at the cost of longer editing times. This kind of versatility of the interface in the second study favored artistic exploration, as reported by subjects and shown in Figures 2.1 and 2.14 and in the supplementary video available in the project page.

Another interesting finding is the fact that all users edited specular highlights as if they were a feature on the surface of the object (Task S3). This is physically inaccurate since they are actually detached from the reflecting surface [211], but nothing was reported during the interviews. This seems to confirm previous findings on the inability of the human visual system to correctly assess the physical accuracy of reflections and highlights [42]. Finally, in terms of workflow and layout, the newly incorporated tools facilitated editing tasks to the point that the right window of the interface was rarely used (see Figure 2.9, Q7). Thus, for the advanced edits shown in the supplementary video and in Figure 2.1, we opted to add a button to open the right window only when needed.

FUTURE WORK   Interfaces for light field editing remain largely unexplored. As such, there are many more opportunities for future studies. We have presented a number of diverse scenes and common tools which we hope help in future research. Additionally, our findings may be applied to edit-

ing RGB+D data, or even stereoscopic editing, which can be regarded as particular cases of light field editing.

However, as with any user study, our conclusions are only strictly valid for the tested scenarios. Exploring other complex interaction procedures, such as selecting a volume in a light field, using edit propagation tools (as sketched in Section 2.7) or applying perspective corrections when pasting objects, are interesting avenues of future work; these would require extending the local, point-based edits used in our experiments to other global or non-local manipulations. Additionally, while our results suggest that users' editing is not significantly affected by depth errors, a quantitative analysis of the effect of imprecise depth remains an open question. Finally, we have explored the two most common paradigms found in literature (*multiview* and *focus*); devising and exploring new paradigms might lead to new interaction workflows for light field editing. It is also important to remark that we have used structured light fields [136], such as those acquired with existing commercial plenoptic cameras (Lytro™, Raytrix™); unconstrained light fields (e.g. [56, 27]) are much less common, and out of the scope of this work.

All the material covered in this paper, including code for the interfaces and to record and analyze data; compiled, functioning versions of all the interfaces; and the raw data itself, are publicly available in the web[7]. We hope that other researchers will be able to build on our work by extending the functionality of the interfaces, sharing some new test light fields or devising new tasks. We believe that our framework and principled evaluation methodology can help gain a better understanding of editing in the multidimensional space of light fields.

APPENDICES

2.A   DESCRIPTION OF TASKS

2.A.1   *Experiment 1: Synthetic Scenarios*

We include here the description given to the users for each *directed* task, while Table 2.2 compiles the task description, task challenge, and central views of the input light field and target image. For open tasks (S6 and S7), target images are given to the users only as a source of inspiration.

TASK S1   *Draw your initial on the back blue wall approximately in the place indicated in the sample image. Use the brush (and the erase tool if necessary). Do not worry about the color of the brush. Time: 5 minutes.*

TASK S2   *Using the brush (and the erase tool if necessary), paint on the pattern of the vase as shown in the sample image to change the color of that part of the vase. Do not worry about the color of the brush. Time: 5 minutes.*

TASK S3   *Using the dodge tool (and the erase tool if necessary), increase the brightness of the specular highlights in the glossy statue of the image. Change only the specular highlights indicated in the sample image. Time: 5 minutes.*

TASK S4   *Once you press Start, an image will appear joined to the cursor. You have to place that image in the scene, so that to appears to be floating in the air. The*

---

7 http://giga.cps.unizar.es/~ajarabo/pubs/lfeiSIG14/
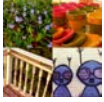
Task Code	Input Central View	Target Central View	Task Description	Task Challenge
S1			Draw initial on wall.	Paint planar geometry parallel to camera plane.
S2			Change the color of the pattern in the vase.	Paint on a curved surface.
S3			Dodge specular highlights in *fertility* figurine.	Modify specular highlights on a curved surface.
S4			Place paper airplane between the vase and the *fertility* figurine.	Place billboard-like object in free space.
S5			Draw a heart on the wall, behind the railing.	Place billboard-like object in free space.
S6			Freely edit the *Head* light field.	Edit a light field with abundant curved surfaces and a very large baseline.
S7			Freely edit the *San Miguel* light field.	Edit a light field with several planar surfaces, parallel and slanted, at very different depths.

Table 2.2: Description of tasks in Experiment 1. See accompanying text for the exact instructions given to users.

image needs to be placed such that in depth it is situated in front of the vase, but behind the glossy statue (see sample image). Time: 5 minutes.

TASK S5   *Using the brush (and the erase tool if necessary) draw, on the back wall, a heart so that it is partially occluded by the railing (see sample image). The heart needs to be on the wall, and thus occluded by the foreground railing. Time: 5 minutes.*

TASK S6   *In this task you can toggle depth information on/off at any point during the editing process. You are given a set of photos for inspiration. Suggestions: painting on the face, adding glasses, monocle, etc. Time: 12 minutes.*

TASK S7   *You can now choose between any of the four interfaces you have tested so far, that is, focus with or without depth, and multiview with or without depth. You can switch between focus and multiview and activate or deactivate depth information at any point during the editing process. The goal is making the scene more beautiful. Suggestions: adding flowers to the plants (the billboard object to insert*

are now some flowers), decorating the flower pots, or any other edit you can think of. Time: 12 minutes.

2.A.2   *Experiment 2: Real Scenarios*

We include here the description given to the users for each task. Table 2.3 compile the task description, task challenge, and central views of the input light field and target image.

TASK R1   *Colorize in green the arrows and time marks of the watch, as shown in the figure. Time: 10 minutes.*

TASK R2   *Two different tasks: (a) Change the color ("Hue" brush) of the nose of the reindeer from brown to red; and (b) change the color ("Hue" brush) of the eyes of the crocodile from white to light yellow. Time: 10 minutes.*

TASK R3   *Place two more street lights on the electric cable, as shown in the image. There is no need to care about the change of size with perspective. You can use the "Paste Img." tool. Time: 10 minutes.*

TASK R4   *Change the color ("Hue" brush) of the large cube statue to dark red to simulate a change of material, as shown in the image. Note: The statue has some holes through which the wall of the building in the back is visible. Time: 10 minutes.*

TASK R5   *Add ivy (using the "Texture" brush) to the wall of the building, as shown in the image. You should try to avoid having ivy on the tree in front of the building and on the pipe on the wall. Time: 10 minutes.*

TASK R6   *Add flowers (use "Paste Img.") to the bush that is behind the railing, taking into account that they should not appear on top of the bars of the railing. Time: 10 minutes.*

TASK R7   *Modify the exposure (you can use the "Dodge" brush) of the matrioska in the foreground to lighten it up, making it less dark. Time: 10 minutes.*

TASK R8   *Add a SIGGRAPH logo (use "Paste Img.") to the spine of the reddish-brown book in the back, and change the color of the blue book (use the "Hue Brush") as shown in the image. Time: 10 minutes.*

TASK R9   *Change the color of the logo in the teapot using the "Hue" brush. Time: 10 minutes.*

TASK R10   *Change the color ("Hue" brush) of the yellow and blue flowers in the foreground to colors in the color range of the rest of the flowers (red, orange, maroon, pink), as shown in the image. Time: 10 minutes.*

2.B   ADDITIONAL DATA FROM ANALYSIS OF EXPERIMENT 1

2.B.1   *Additional Information on Experimental Procedure*

The study consisted of two main blocks: *multiview* and *focus*, in randomized order for each user. Within each block, the two versions of the interface were used (with and without depth), also in randomized order to compensate for
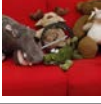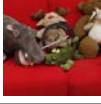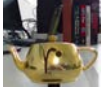
Task Code	Input Central View	Target Central View	Task Description	Task Challenge
R1			Colorize the arrows and the marks on the inner circumference of the watch.	Painting on a slanted surface with a noisy depth reconstruction. *Depth from Wanner and Goldlücke [230].*
R2			Change the color of the reindeer's nose and of the eyes of the crocodile.	Painting on a curved surface. *Depth from Kim et al. [117].*
R3			Place another light on the cable.	Placing a billboard-like object in free space. *Depth from Kim et al. [117].*
R4			Change the hue of the statue.	Dealing with selection and complex geometries. *Depth from Kim et al. [117].*
R5			Add ivy to the wall, as shown in the image.	Dealing with occlusions and slanted surfaces. *Depth from Kim et al. [117].*
R6			Add flowers to the bush (see target image for guidance).	Dealing with occlusions and working on areas of complex depth reconstruction. *Depth from Kim et al. [117].*
R7			Use dodge to brighten-up the matrioska in the foreground.	Editing curved surfaces with a coarse depth reconstruction. *Depth from Lytro^{TM} [145].*
R8			Change one of the books' color and paste a SIGGRAPH logo on a book.	Dealing with color selection and pasting onto an object parallel to the camera plane. *Depth from Lytro^{TM} [145].*
R9			Change the RenderMan logo in the teapot to a purplish color.	Painting on a curved surface with a coarse depth reconstruction. *Depth from Lytro^{TM} [145].*
R10			Change the colors of the two foremost flowers in the scene to match those of the rest of the flowers.	Dealing with selection of complex geometries. Using selection based on color and/or on depth. *Depth from Lytro^{TM} [145].*

Table 2.3: Description of tasks in Experiment 2. See accompanying text for the exact instructions given to users.

possible learning effects. This yielded a total of four sessions, with each one including all five tasks sequentially (S1 to S5). After each block, subjects

were asked to complete Task S6 with the current interaction paradigm. After completing both blocks, subjects additionally performed the final Task S7. We recorded the screen during all the experiments.

After finishing a session with an interface or an open task, users had to fill in a questionnaire and could write free-form comments as well. At the end, subjects were required to fill in a final questionnaire where they had to rate and rank interfaces per task, and also regarding other more general aspects. All questionnaires can be found in the supplementary document[8] at the project page. Each participant completed the experiment with an informal interview, to collect general impressions and ask about the subject's workflow.

Although the participants were recommended to use a pen on a tablet, they were allowed to use a mouse if they felt more comfortable using it, to ensure that their performance was not affected by the input device. The full experiment took around four hours per subject, including training and short breaks. The training took around one hour, including filling in a preliminary questionnaire, and was performed with an additional light field, shown in the supplementary material.

2.B.2 *Error in Depth*

Here, Table 2.4 shows pairwise comparisons (*p*-value) for the error in depth of each of the five *directed* tasks [S1..S5]. For the results of the ANOVA see also Table 1 in the main text. A *p*-value ≤ 0.05 (marked with a star (*)) indicates the difference between interfaces is significant. Additionally, in Figure 2.15 we plot 95% confidence intervals for the difference of the mean between each pair of interfaces. Confidence intervals also show significance (if the interval contains zero, then the difference between the compared interfaces is not significant), but additionally they give an idea of the magnitude of the difference. Since confidence intervals are symmetric for each pair of interfaces (e.g. between $M - F$ and $F - M$ only the sign of the interval changes) we only show half of the pairwise comparisons.

2.B.3 *Time to Completion*

We plot mean times to completion per interface for each directed task (S1–S5) in Figure 2.16, and also illustrate in it statistically significant differences between them. For tasks S1 to S3, which require placing strokes on surfaces, interfaces with depth information (*MD* and *FD*) took less time, although the difference is only significant with respect to *M* ($p \leq 0.008$). There is no significant differences in Task S1, due to its simplicity.

We additionally provide here, in Table 2.5, the results of the repeated measures ANOVA performed on the time to completion, from which the main text only reports significant differences. The table contains the *H*-test, the between-groups degrees of freedom df_1 (three unless the Greenhouse-Geisser correction is applied because sphericity is violated), the within-groups degrees of freedom df_2, the associated *p*-value, and the value of the partial eta-squared η^2 for each task, indicative of the proportion of variance that can be attributed to the *interface* factor. Table 2.6 contains the pairwise comparisons (*p*-value) for the time to completion in each of the five *directed* tasks [S1..S5]. A *p*-value ≤ 0.05 (marked with *) indicates significant differ-

---

8 http://giga.cps.unizar.es/~ajarabo/pubs/lfeiSIG14/downloads/Jarabo_sig14_Supplementary_Material.pdf

Table 2.4: Significance of pairwise comparisons for error in depth in directed tasks.

	M	MD	F	FD
M	-	0.000*	0.018*	0.000*
MD	0.000*	-	0.000*	-
F	0.018*	0.000*	-	0.000*
FD	0.000*	-	0.000*	-

(a) Task S1

	M	MD	F	FD
M	-	0.000*	0.000*	0.000*
MD	0.000*	-	0.000*	-
F	0.000*	0.000*	-	0.000*
FD	0.000*	-	0.000*	-

(b) Task S2

	M	MD	F	FD
M	-	0.000*	0.000*	0.000*
MD	0.000*	-	0.000*	-
F	0.000*	0.000*	-	0.000*
FD	0.000*	-	0.000*	-

(c) Task S3

	M	MD	F	FD
M	-	0.000*	0.951	0.000*
MD	0.000*	-	0.000*	0.296
F	0.951	0.000*	-	0.000*
FD	0.000*	0.296	0.000*	-

(d) Task S4

	M	MD	F	FD
M	-	0.018*	0.014*	0.028*
MD	0.018*	-	0.606	0.285
F	0.014*	0.606	-	0.024*
FD	0.028*	0.285	0.024*	-

(e) Task S5

ence. Additionally, in Figure 2.17 we plot 95% confidence intervals for the difference of the mean between each pair of interfaces (see Section 2.B.2 for details on confidence intervals).

Table 2.5: ANOVA results for time to completion in directed tasks.

	S1	S2	S3	S4	S5
H	2.048	6.730	5.431	3.986	9.175
(df_1, df_2)	(2.080,35.364)	(3,54)	(1.815,32.669)	(3,48)	(3,54)
p	0.142	0.001*	0.011*	0.013*	0.000*
η^2 (%)	10.8	27.2	23.2	19.9	33.8

2.B.4 *Ratings*

Users were asked to rate their preferences in *directed* tasks (S1..S5), overall preference, and general aspects on a scale [1..5]. Mean ratings for directed tasks and for overall preference can be found in Figure 2.18, while in Figure 2.19 we show the mean values for the questions on general aspects, as well as the results of the pairwise comparisons between interfaces (for both ratings and rankings for comparison purposes).

Next we provide the results of the repeated measures ANOVA performed on the ratings, from which the main text only reports which differences between interfaces are significant. Table 2.7 provides the *H*-test, degrees of freedom, and its associated significance *p*. The between-groups degrees of freedom are three in all cases, since we have four interfaces and sphericity can be assumed, while the within-group degrees of freedom are 57 in all

(a) Task S1   (b) Task S2   (c) Task S3

(d) Task S4   (e) Task S5
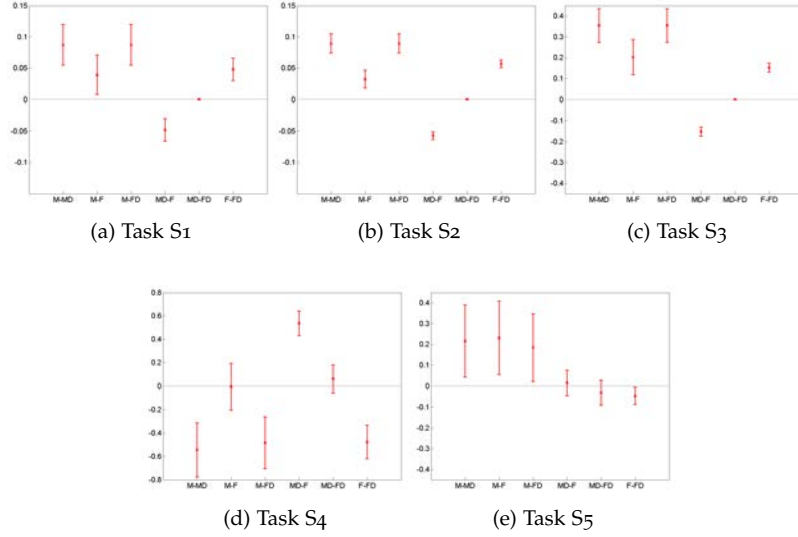
Figure 2.15: Confidence intervals at 95% for mean difference of error in depth between interfaces for Tasks 1 to 5.
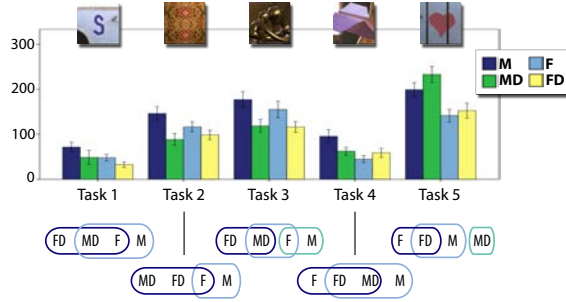
Figure 2.16: Top: Mean time to completion per interface for each task. Bottom: Pairwise comparisons for the time to completion in each task. Items in the same set are statistically indistinguishable.

cases. Additionally, we include the partial eta-squared η^2 for each case, and the significance results (p-value) of the pairwise comparisons in Table 2.8 (we found no significant difference for *accuracy*, see Table 2.7). A p-value ≤ 0.05 (marked with *) indicates the difference between interfaces is significant.

2.B.5   *Rankings*

Similarly, users ranked preferences in directed tasks (S1..S5), overall preference, and general aspects. Rankings for preferences in directed tasks and for overall preference can be found in the main text (Figure 6, while here in Figure 2.20 we show the ranks for the questions on general aspects. Additionally, results of pairwise comparisons between interfaces for rankings on general aspects questions are shown in Figure 2.19.

Users' preferences for the questions on general aspects (both in rankings and ratings, see Figure 2.19, in addition to Figure 2.20) show that *F* ranks first in most cases, with no significant difference among the others. When it comes to accuracy, agreement among users decreases, and differences turn

Table 2.6: Significance of pairwise comparisons for time to completion in directed tasks.

	M	MD	F	FD
M	-	0.293	0.104	0.007*
MD	0.293	-	0.977	0.367
F	0.104	0.977	-	0.115
FD	0.007*	0.367	0.115	-

(a) Task S1

	M	MD	F	FD
M	-	0.001*	0.093	0.006*
MD	0.001*	-	0.061	0.402
F	0.093	0.061	-	0.062
FD	0.006*	0.402	0.062	-

(b) Task S2

	M	MD	F	FD
M	-	0.002*	0.386	0.008*
MD	0.002*	-	0.063	0.850
F	0.386	0.063	-	0.004*
FD	0.008*	0.850	0.004*	-

(c) Task S3

	M	MD	F	FD
M	-	0.056	0.008*	0.068
MD	0.056	-	0.131	0.814
F	0.008*	0.131	-	0.294
FD	0.068	0.814	0.294	-

(d) Task S4

	M	MD	F	FD
M	-	0.050*	0.003*	0.052
MD	0.050*	-	0.000*	0.004*
F	0.003*	0.000*	-	0.590
FD	0.052	0.004*	0.590	-

(e) Task S5

out not significant. Overall, what we extract from this analysis is users' inclination towards the *focus without depth* interface. The high correlation between rankings and ratings is apparent.

We provide here as well the results of the Kruskal-Wallis test performed on the rankings, from which the main text only reports which differences between interfaces are significant. Table 2.9 provides the test statistic χ^2, its degrees of freedom (three in all cases, since we have four interfaces) and its associated significance p. We also include the significance results of the pairwise comparisons in Table 2.10 (we found no significant difference for *accuracy*, see Table 2.9). A p-value ≤ 0.05 (marked *) indicates the difference between interfaces is significant.

For each ranking obtained in each question, we obtain the rank product per interface $\Psi(\vartheta)$ (see main text for details on computation). This rank product is used when sorting the interfaces according to the rankings received. In Table 2.11 we include all the rank products per interface per question, highlighting in bold the highest ranked.

2.B.6 *Workflow in open tasks*

Figure 2.21 shows usage times for each interface and each of the open tasks, divided into the action being performed in each (drawing, erasing, changing the view or adjusting depth). The reader may refer to the supplementary videos[9] for sample editing sessions by subjects for these tasks.

---

9 http://giga.cps.unizar.es/~ajarabo/pubs/lfeiSIG14/videos/Jarabo2014_suppvideo_
Experiment1.mp4
http://giga.cps.unizar.es/~ajarabo/pubs/lfeiSIG14/videos/Jarabo2014_suppvideo_
Experiment2.mp4

(a) Task S1   (b) Task S2   (c) Task S3

(d) Task S4   (e) Task S5
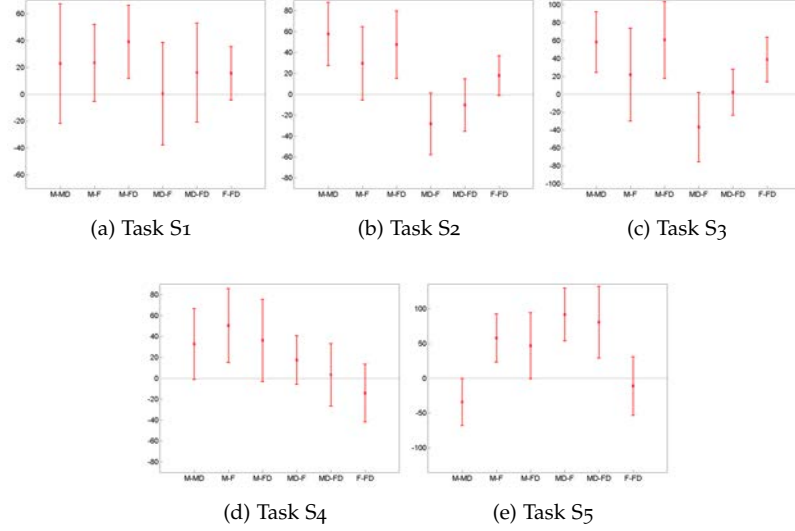
Figure 2.17: Confidence intervals at 95% for mean difference in time to completion between interfaces for Tasks S1 to S5.
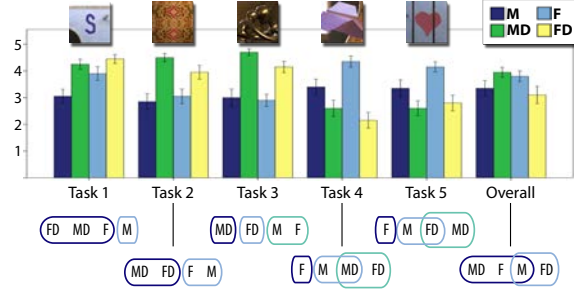
Figure 2.18: Top: Mean ratings from final questionnaire for questions on preference for each task and overall preference. Bottom: Pairwise comparisons between interfaces for the ratings. Items in the same set are statistically indistinguishable.

In Task S6, the times spent with and without depth for each interface are relatively balanced. This situation changes in Task S7, possibly as a consequence of the different nature of the light fields involved: the *head* in Task S6 is a large non-planar surface, where having depth information is highly useful, whereas *San Miguel* in Task S7 has many flat surfaces and larger depth discontinuities with free-space in between.

In Table 2.12 we show the number of times users switched from one interface to another in Task S7, in which they can freely switch between any of the four interfaces at any time. We show the sum for all subjects. Note that, due to how menus were implemented, users did not select one of four interfaces, but switched between *multiview* and *focus* paradigms, and between depth on or off (there are eight possible interface switches). The high number of switches between *M* and *F* supports the findings reported in the main text: the preferred workflow was to edit mostly in *F*, then switch to *M* for visualization.
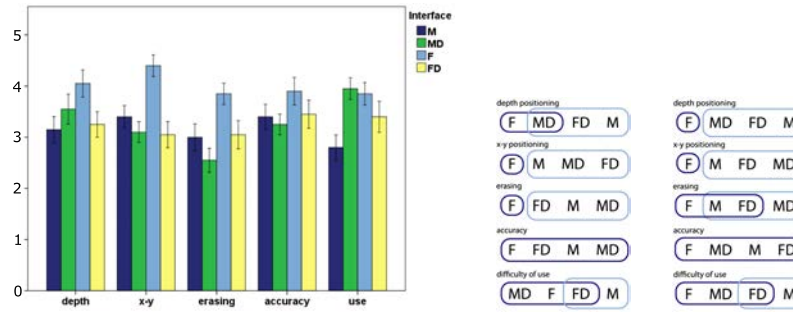
Figure 2.19: From left to right, for final questions on general aspects: mean ratings for each interface; ratings ordered by mean; and rankings ordered by rank product. Groupings show significant differences between interfaces.

Table 2.7: ANOVA results for ratings in final questionnaire.

	S_1	S_2	S_3	S_4	S_5	*overall*
$H(3,57)$	7.410	9.251	13.203	12.390	6.218	2.217
p	0.000*	0.000*	0.000*	0.000*	0.001*	0.096
η^2 (%)	28.1	32.7	41.0	39.5	24.7	10.4

	depth	*x-y*	*erasing*	*accuracy*	*difficulty*
$H(3,57)$	2.053	8.456	4.180	1.119	3.943
p	0.117	0.000*	0.010*	0.349	0.013*
η^2 (%)	9.8	30.8	18.0	5.6	17.2

2.C ADDITIONAL DATA FROM ANALYSIS OF EXPERIMENT 2

2.C.1 *Distribution of Times*

Figure 2.23 illustrate the average times spent by users doing different actions and using different tools or features, namely: adjusting depth, changing view, drawing, erasing, setting the depth threshold (for the active depth range), setting the tool size, using the Color Selection tool, using the Depth Selection tool, and with the Visual Aid activated, plus the total time. For each of these actions or tools, the times spent in each of the four interface configurations (multiview without depth - M, focus without depth - F, multiview with depth - MD, and focus with depth - FD), as well as the total time, is shown.
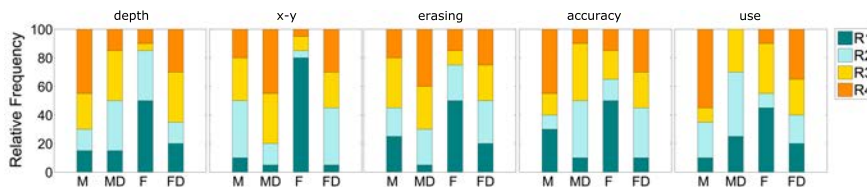
Figure 2.20: Rankings for each interface for questions on general aspects asked in final questionnaire.

Table 2.8: Significance of pairwise comparisons for ratings in final questionnaire.

	M	MD	F	FD
M	-	0.000*	0.025*	0.001*
MD	0.000*	-	0.384	0.494
F	0.025*	0.384	-	0.053
FD	0.001*	0.494	0.053	-

(a) Task S1

	M	MD	F	FD
M	-	0.000*	0.541	0.019*
MD	0.000*	-	0.001*	0.077
F	0.541	0.001*	-	0.016*
FD	0.019*	0.077	0.016*	-

(b) Task S2

	M	MD	F	FD
M	-	0.000*	0.815	0.014*
MD	0.000*	-	0.000*	0.045*
F	0.815	0.000*	-	0.000*
FD	0.014*	0.045*	0.000*	-

(c) Task S3

	M	MD	F	FD
M	-	0.053	0.026*	0.014*
MD	0.053	-	0.000*	0.107
F	0.026*	0.000*	-	0.000*
FD	0.014*	0.107	0.000*	-

(d) Task S4

	M	MD	F	FD
M	-	0.036*	0.035*	0.270
MD	0.036*	-	0.001*	0.618
F	0.035*	0.001*	-	0.002*
FD	0.270	0.618	0.002*	-

(e) Task S5

	M	MD	F	FD
M	-	0.131	0.275	0.566
MD	0.131	-	0.614	0.047*
F	0.275	0.614	-	0.044*
FD	0.566	0.047*	0.044*	-

(f) Overall

	M	MD	F	FD
M	-	0.338	0.041*	0.823
MD	0.338	-	0.220	0.410
F	0.041*	0.220	-	0.049*
FD	0.823	0.410	0.049*	-

(g) Depth Positioning

	M	MD	F	FD
M	-	0.316	0.001*	0.309
MD	0.316	-	0.001*	0.871
F	0.001*	0.001*	-	0.000*
FD	0.309	0.871	0.000*	-

(h) x-y Positioning

	M	MD	F	FD
M	-	0.154	0.034*	0.910
MD	0.154	-	0.003*	0.234
F	0.034*	0.003*	-	0.022*
FD	0.910	0.234	0.022*	-

(i) Erasing

	M	MD	F	FD
M	-	0.002*	0.015*	0.219
MD	0.002*	-	0.785	0.102
F	0.015*	0.785	-	0.216
FD	0.219	0.102	0.216	-

(j) Difficulty of Use

Table 2.9: Kruskal-Wallis results for rankings in final questionnaire.

	S1	S2	S3	S4	S5	overall
$\chi^2(3)$	26.149	14.931	35.313	22.357	11.455	9.006
p	0.000*	0.001*	0.000*	0.000*	0.008*	0.028*

	depth	x-y	erasing	accuracy	difficulty
$\chi^2(3)$	13.825	28.440	10.507	5.925	12.403
p	0.002*	0.000*	0.014*	0.116	0.005*

Table 2.10: Significance of pairwise comparisons for rankings in final questionnaire.

	M	MD	F	FD
M	-	0.000*	0.025*	0.000*
MD	0.000*	-	0.206	0.160
F	0.025*	0.206	-	0.008*
FD	0.000*	0.160	0.008*	-

(a) Task S1

	M	MD	F	FD
M	-	0.001*	0.673	0.011*
MD	0.001*	-	0.005*	0.482
F	0.673	0.005*	-	0.035*
FD	0.011*	0.482	0.035*	-

(b) Task S2

	M	MD	F	FD
M	-	0.000*	0.482	0.002*
MD	0.000*	-	0.000*	0.206
F	0.482	0.000*	-	0.000*
FD	0.002*	0.206	0.000*	-

(c) Task S3

	M	MD	F	FD
M	-	0.122	0.035*	0.025*
MD	0.122	-	0.000*	0.482
F	0.035*	0.000*	-	0.000*
FD	0.025*	0.482	0.000*	-

(d) Task S4

	M	MD	F	FD
M	-	0.122	0.122	0.261
MD	0.122	-	0.002*	0.673
F	0.122	0.002*	-	0.008*
FD	0.261	0.673	0.008*	-

(e) Task S5

	M	MD	F	FD
M	-	0.011*	0.160	1.000
MD	0.011*	-	0.261	0.011*
F	0.160	0.261	-	0.160
FD	1.000	0.011*	0.160	-

(f) Overall

	M	MD	F	FD
M	-	0.160	0.000*	0.482
MD	0.160	-	0.035*	0.482
F	0.000*	0.035*	-	0.005*
FD	0.482	0.482	0.005*	-

(g) Depth Positioning

	M	MD	F	FD
M	-	0.092	0.001*	0.574
MD	0.092	-	0.000*	0.261
F	0.001*	0.000*	-	0.000*
FD	0.574	0.261	0.000*	-

(h) x-y Positioning

	M	MD	F	FD
M	-	0.122	0.092	0.888
MD	0.122	-	0.001*	0.160
F	0.092	0.001*	-	0.068
FD	0.888	0.160	0.068	-

(i) Erasing

	M	MD	F	FD
M	-	0.003*	0.005*	0.325
MD	0.003*	-	0.888	0.049*
F	0.005*	0.888	-	0.068
FD	0.325	0.049*	0.068	-

(j) Difficulty of Use

Table 2.11: Rank products per interface for rank scores on final questionnaire.

	S1	S2	S3	S4	S5	overall
M	3.37	2.85	2.90	2.16	2.17	2.59
MD	1.99	**1.66**	**1.39**	2.72	2.70	**1.71**
F	2.34	2.66	3.28	**1.37**	**1.64**	2.07
FD	**1.52**	1.90	1.81	2.98	2.49	2.62

	depth	x-y	erasing	accuracy	difficulty
M	2.72	2.42	2.23	2.36	2.85
MD	2.30	3.04	2.88	2.35	1.90
F	**1.55**	**1.24**	**1.63**	**1.70**	**1.81**
FD	2.47	2.63	2.29	2.54	2.46
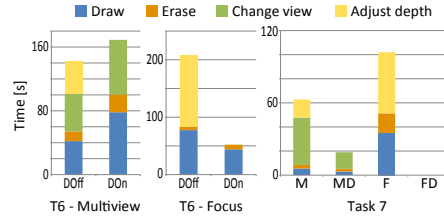
Figure 2.21: From left to right: Distribution of times for Task S6 using the *multiview* and *focus* paradigms, and for Task S7. Note that we do not take idle times into account. We plot median values, which makes *FD* in Task S7 become zero in all four categories.

Table 2.12: Switching between interfaces in Task S7.

	F→M	F→FD	M→MD	FD→MD
$N_{switches}$	58	29	25	18
	M→F	FD→F	MD→M	MD→FD
$N_{switches}$	52	27	15	31

2.C.2 *Rankings of Difficulty*

In the final questionnaire, we ask users to rank the tasks in difficulty, 10 being the easiest and 1 the most difficult. Figure 2.22 shows the rankings and corresponding rank products for all tasks, computed as explained in Section 2.5.
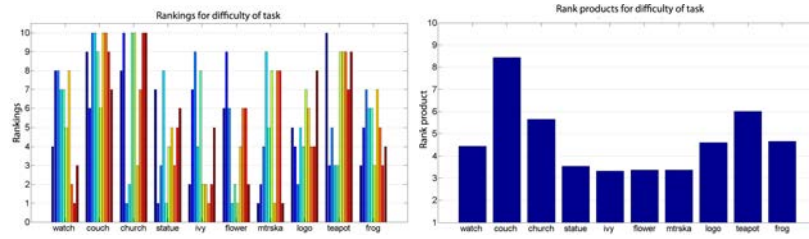
Figure 2.22: Rankings (left) and Rank products (right) for difficulty for each task, all users.
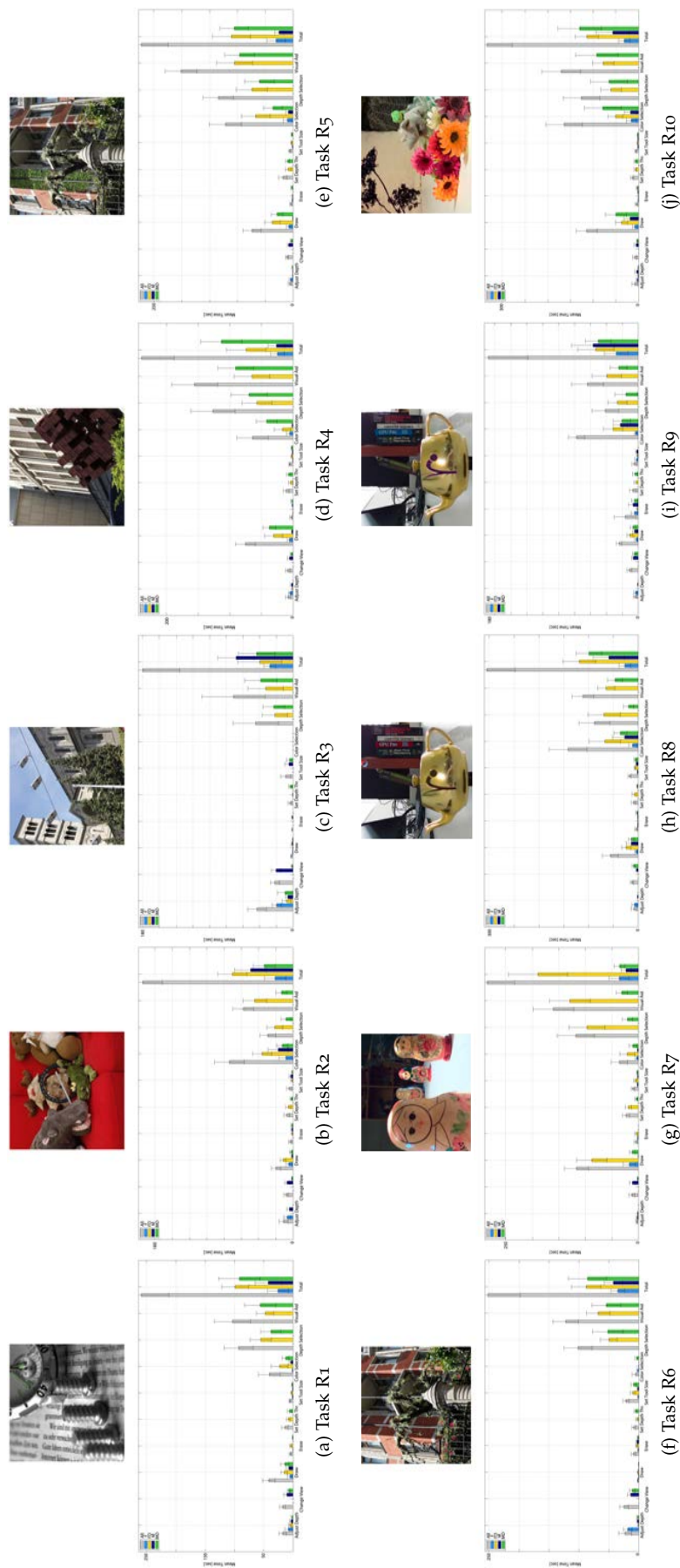
Figure 2.23: Times spent doing different actions and using different tools or features in Tasks R1 to R10, with each of the four interface configurations and in total. Please refer to text for more details.

EFFECTS OF APPROXIMATE FILTERING ON THE APPEARANCE OF BIDIRECTIONAL TEXTURE FUNCTIONS

In this chapter we present the first systematic study of the effects of approximate filtering on the appearance of Bidirectional Texture Functions (BTF), that define the appearance of complex surfaces in a spatio-bidirectional domain. We perform a variety of perceptual experiments exploring the spatial, angular and temporal domains over a varied set of stimuli, on both simple and complex geometry and lighting conditions. Then, we correlate our findings with low- and high-level descriptors of the BTF, and show different practical applications of our findings in filtering, rendering or BTF compression. This work has been published in *IEEE Transactions on Visualization and Computer Graphics*, and presented at *Pacific Graphics 2014*.

3.1 INTRODUCTION

Many computer graphics applications require accurate depiction of visually rich material appearance. Bidirectional Texture Functions (BTFs) represent complex spatially and angularly-varying appearance, including effects such as self-shadowing, inter-reflections or subsurface scattering. BTFs are usually captured by taking photographs of a material sample under different combinations of light and view directions.

Filtering is required to render BTFs without aliasing. Mathematically exact filtering is infeasible with finite processing power and/or memory. Approximate filtering techniques are needed, with the acceptability of approximations measured by their perceptual impact. In this paper we present the first systematic study of the effects of multi-dimensional filtering on the appearance of BTFs. Our goal is to understand how approximate filtering of BTFs along the spatial, angular and temporal dimensions (both for a moving light source and a moving camera) affects the perceived visual quality of the results (see Figure 3.1).

Previous studies have focused on developing efficient *compression* algorithms (see [156, 41]). Recently, Filip et al. showed that different materials actually require different compression strategies [39]. Another related line of work deals with *filtering* strategies, where pre-computations are used for efficient rendering, by computing multiple pre-filtered representation of appearance at different viewing distances. Given the sophisticated, non-linear illumination effects stored in a BTF, relatively complex filters are employed [13], although visual appearance is usually not taken into account.

Our work complements these two lines of research: We study how different filtering strategies affect the perceived appearance of a varied set of BTFs, by means of systematic psychophysical experiments. We use a var-
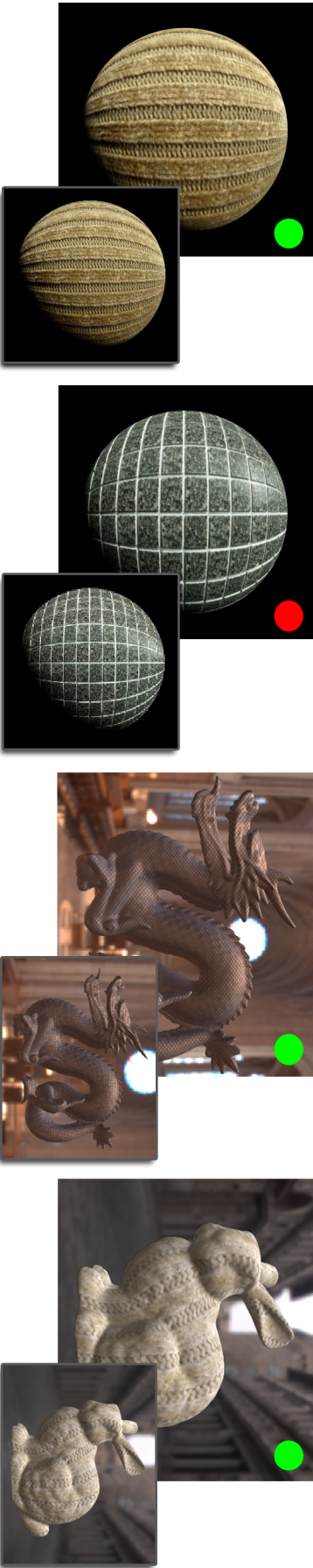
Figure 3.1: We perform psychophysical experiments on simple geometries to examine filtering strategies for BTF rendering. From left to right: (1) Our experiments reveal that the filtered, aggressively undersampled representation (smaller image) of the BTF is perceived as visually equivalent to its reference, multi sampled version (larger image). (2) However, the same experiments show that the same strategy on the second sphere is not considered visually equivalent. We apply our observations to produce filtered examples of more complex scenes (3 and 4) which experiments show maintain visual equivalence.

ied subset of the Bonn and UCSD BTF databases, and analyze them both globally and according to high-level descriptors of their visual properties.

CONTRIBUTIONS   Our main findings are:

- Approximate pre-filtered representations of BTFs can be used without affecting visual equivalence with a multi-sampled reference solution

- For static scenes, there is a great tolerance for aliasing in the spatial and angular domains, which is preferred to over-blurred BTFs. For dynamic scenes, this trend is reversed, and over-blurred animations are preferred over temporal aliasing artifacts. These findings correlate well with known mechanisms of human perception

- The angular domain can be more aggressively filtered, while filtering in the spatial domain rapidly affects visual equivalence

- High-level descriptors of the BTFs (such as e.g. *glossy* or *structured*) correlate well with perceptually equivalent levels of filtering. This suggests the necessity and usefulness of standardized high-level descriptors in material and BTF databases. In turn, these high-level descriptors are also correlated with low-level statistics of the BTF

- We show that our findings generalize to different geometries and illumination conditions, and propose different practical applications in filtering, rendering and compression strategies

This is the first work to systematically analyze the perceptual effect on material appearance of different approximate BTF filtering strategies. While we did not consider all possible combinations of all parameters, we believe that our findings should provide enough traction to motivate future work, for have made our stimuli and data publicly available[1]. We hope that this work can also inspire future compression and filtering strategies for BTFs, which in turn may lead to more sophisticated rendering and editing algorithms.

3.2   PREVIOUS WORK

BTF COMPRESSION   Dana et al. [25] first introduced BTF as an image-based representation for material appearance. Müller et al. [156] presented a comprehensive survey on BTF acquisition, synthesis and rendering. Several BTF compression methods were compared, but none of them took into account the perception of BTF at different viewing distances or under different light directions. Another excellent survey including subsequent work in the field can be found in [40].

More recently, Ruiters et al. [192] achieved high compression rates for single-level BTFs, by fitting a small set of basis functions based on tensor decomposition. Tsai et al. [215] further pushes the idea to a k-clustered tensor approximation, which is suitable for efficient real-time rendering of the compressed BTFs. Havran et al. [75] compressed the BTF by adopting a multi-dimensional conditional probability density function in conjunction with vector quantization. Mipmapping was handled by directly applying the same algorithm to averaged BTF data. In contrast to our work, no perceptual factors were considered.

---

1 http://giga.cps.unizar.es/~ajarabo/pubs/btfTVCG14/index.html

HIERARCHICAL REPRESENTATION OF APPEARANCE   Various algorithms have been proposed to pre-filter material appearance, focusing on texture maps [244], normal maps [210, 72], or general geometry and BRDFs [248]. A survey for reflectance filtering algorithms was recently presented by Bruneton and Neyret [13]. Ma et al. [146] pre-filtered BTFs using Principal Component Analysis (PCA) and parametric fitting of Phong models after a Laplacian transformation. Impressive real-time LOD rendering results were achieved. However, their method is purely numerical, and no evaluation of how the BTF appearance is perceived for different levels of detail is provided.

PERCEPTION   The study of visual perception in the context of computer graphics has produced many useful results [152]. Some example applications include selective rendering [207], global illumination [159, 104] and motion blur [163]. Rogowitz and Rushmeier [190] found that simplified geometries were perceived differently for static and animated cases. Ramanarayanan et al. [183] and Vangorp et al. [217] studied how object geometry, material, and illumination interact to provide information about appearance. Krivanek et al. [131] investigated the relationship between parameters of the virtual point lights algorithm with the perception of different materials.

Meseth et al. [153] evaluated the rendering quality obtained by using BTFs instead of simple 2D textures. A perception-based metric for single-level BTFs was derived in [60], in order to achieve higher compression rates. Filip et al. [39, 40] ran psychophysical tests to discard perceptually unimportant data in the BTF. Although closer to our approach, our work differs in several ways: First, we do not aim at reducing the input dataset of 2D texture images that define the BTF; instead, we focus on efficient strategies to sample such multidimensional data, and on studying the effect of pre-filtering BTFs in their perceived appearance. Second, Filip et al. use a statistical description of the BTFs, whereas we rely on more intuitive, high-level material descriptors. Third, they compare results based on per-pixel visible differences between the compressed and the original data, which is not related to higher-level visual properties; instead, our work relies on the concept of *visual equivalence*, where visibly different images are considered equivalent if they convey the same impression of appearance. Our work offers applications beyond data compression, ranging from level-of-detail or filtering techniques for BTFs, to optimized rendering. Our findings could be applied in conjunction with compression techniques, resulting in good filtering of reduced data.

3.3   PROBLEM STATEMENT

FILTERING A BTF   Rendering a richly textured surface at different distances without aliasing artifacts requires high sampling rates at considerable computational cost. To address this issue, one common approach is to pre-filter the appearance of the surface [13]. However, to obtain the exact solution, we would need to compute the filtered appearance for all light and view conditions. This is impractical, due to huge storage and computation requirements. Instead, the filtered appearance is typically approximated, by pre-filtering only a subset of viewing distances, and then building a hierarchy of level-of-detail (LOD) representations.

Texture *mipmapping* is one such technique to reduce aliasing for surfaces decorated with 2D plain textures. This technique creates a hierarchy of different versions of the texture, where each level is a down-sampled version of the previous level. When the surface is rendered, the level is chosen so that
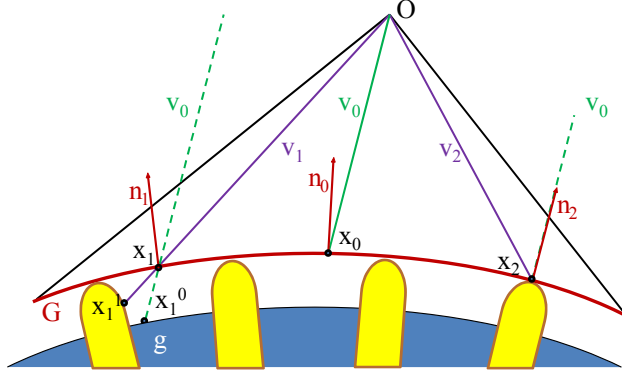
Figure 3.2: This diagram shows the geometry of filtering the reflectance of surface G (red, representing the area covered in one pixel). Appearance is modeled with a BTF describing the underlying meso-geometry g (blue-yellow surface). O denotes the viewpoint (please note that we have exaggerated its proximity to the pixel for illustration purposes), while v_0 (green) is the viewing direction (vector $O - x_0$), assumed to be constant. Sampling points x_1 and x_2 from the constant or from the correct view directions will yield different results. Note that even using an orthogonal projection (i.e. constant viewing direction v_0), the angle between the viewing direction and the normal at the differential points in the surface varies according to the surface's curvature, which creates an effective non-zero solid angle.

the ratio $r = t{:}p$ (texel to pixel) is preserved at 1:1. However, it is difficult to extend texture mipmapping to BTFs, since it makes assumptions about the underlying surface geometry, which may not hold for complex surface representations as BTFs. For example, one assumption is that the normal n of the surface and the light l and view v directions are uniform inside the fraction of the surface contained in each pixel. This is not valid for BTFs, as four additional directional dimensions defining l and v need to be sampled, to take into account the effect of the underlying meso-geometry.

Figure 3.2 shows this. The red line represents the portion of geometry G covered by one pixel, which has an underlying meso-structure g (the blue and yellow shape) modeled with a BTF. The resulting outgoing radiance of the pixel is the integral of the reflected light. However, this constant view vector will introduce errors, due to parallax and to the directional dependence of the reflectance. For point x_1 in G, the sampled point in g would be $g(x_1, v_0) = x_1^0$, yielding a wrong blue sample. Using the correct view direction v_1 we obtain $g(x_1, v_1) = x_1^1$, a correct yellow sample (parallax problem). For x_2, the sampled reflected light is likely to be different for both view directions; this is particularly problematic for glossy surfaces.

Similar problems occur when the incoming light direction l is assumed to be constant. This means that, in practice, a correct BTF filtering should not only preserve a correct ratio $t{:}p$ in the spatial domain, but also filter the view v and light l directions correctly.

PERCEPTUAL CONSEQUENCES OF FILTERING BTF   While necessary to avoid the artifacts described above, approximately pre-filtering the BTF might produce new, different artifacts: if the filter size is too small, aliasing artifacts might appear on the surface. On the other hand, if the filter is too large, the appearance of the surface will get blurred, decreasing overall contrast and detail.

Our main goal is to evaluate under which conditions an approximately pre-filtered BTF is considered *visually equivalent* [183, 131] to the ground-truth image. We want to explore how different kernel sizes applied on the spatial, directional and temporal domains of the BTF affect the appearance of the surface at different viewing and lighting conditions, both for static images and animation. Particularly, we are interested in the following questions:

- Is it possible to approximately pre-filter BTFs while maintaining visual equivalence with a multi-sampled reference?

- Is this perceived equivalence correlated with high-level visual properties of the surface?

- What kind of artifacts (e.g. aliasing, blur) are more easily accepted by the human visual system? Under what conditions?

- Can different sampling strategies be applied to the different domains of the BTF? What is the interplay between domains?

- How do different filtering kernels affect visual equivalence?

- Does the distance to the camera (i.e. projected area in the pixel) affect visual equivalence?

- Does the motion of camera and light sources affect visual equivalence?

3.4 OVERVIEW OF THE EXPERIMENTS

By applying different filtering kernels on the domains of the BTF (spatial and angular), we want to analyze how each domain affects the appearance of the filtered BTFs, as well as the interplay between them. We have designed three different experiments: The first one evaluates the perception of filtered BTFs on static images, seen at different distances and under different illumination directions. The second one analyzes the effect of a varying illumination vector l, while the third analyzes the effect for variations in the view vector v.

For all the experiments, we use a simple scene (rendered at 512×512 pixels) consisting of a sphere, with material appearance modeled using a BTF, illuminated by a single directional light. We choose this setup following the work by Filip et al. [39], whose results suggest that simple geometry with directional illumination is less forgiving than more complex geometry and illumination. Vangorp et al. [217] found that simple objects like an sphere are actually not appropriate for depicting material. However, the authors only explore reflectance and explicitly leave spatially varying effects (e.g. textures or BTFs) as a future line of research. On the contrary, the work by Filip et al. focuses on the particular case of BTF's, so their findings are better suited for our research. Generalization to different object shapes and illuminations is later evaluated in Section 3.8.

We describe here common aspects of all the experiments. We also show how our multidimensional filtering strategy has advantages over classic (spatial) mipmapping by means of a pilot study. Last, we give specific details of each experiment in subsequent sections.
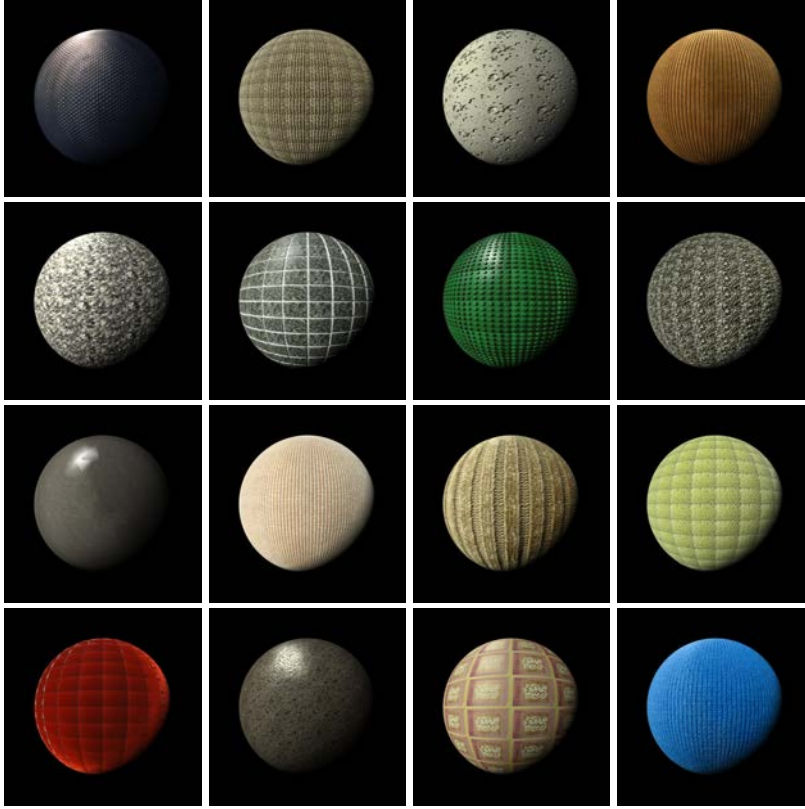
Figure 3.3: Reference renderings of the stimulus BTFs. From top to bottom, and from left to right: *Cambrils*, *Carpet*, *Ceiling*, *Corduroy*; *Floortile*, *Impala*, *Lego*, *Lichen*; *Pinktile*, *Proposte*, *Pulli*, *Sponge*; *Velvet*, *Walkway*, *Wallpaper* and *Wool*.

INPUT BTFS   We use the Bonn [195] and UCSD [127] databases. These BTFs present a low to moderate angular frequency, restricted by the capabilities of the acquisition devices. This in turn limits the types of materials that can be represented using BTFs. Therefore, we limit our study to the materials that BTFs excel at representing. We use captured BTFs since they are the most common and used in practice. We visually analyzed high-resolution multisampled renderings of several BTFs in order to detect pixel-wise registration inaccuracies, and discarded unsuitable ones. In the end, we keep sixteen different BTFs (see Figure 3.3). Each one is made up of up to 151×151 images with a resolution up to 256^2 pixels. They have been chosen to represent a wide range of surfaces, showing different levels of complexity, both in reflectance and meso-geometry. We use uncompressed versions of the BTF, to avoid artifacts. Since these BTFs represent very different materials, we classify them according to a set of high level properties describing their appearance, by means of a pilot study performed by ten participants. These properties are based on previous work on texture [184] and BRDF [151] classification. The purpose of this descriptor-based classification is to analyze the results both globally, and at descriptor level. Table 3.1 shows the result of the pilot study. More details on the pilot study can be found in Section 3.A.

FILTERING KERNELS   We test three different filters in our experiments: *box*, *Gaussian* and *Lanczos*. The box filter is widely used, since it is the sim-

BTF	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)	(15)
Cambrils	○	○	●	○	●	○	○	●	○	●	●	○	○	○	○
Carpet	○	○	●	●	●	○	○	●	○	○	○	●	○	○	○
Ceiling	○	○	○	●	○	●	○	●	●	○	○	○	●	○	●
Corduroy	○	○	●	○	○	○	○	●	○	●	○	●	●	○	○
Floortile	○	○	○	○	●	●	○	○	○	○	○	○	○	○	○
Impala	●	○	●	○	●	○	○	●	●	○	○	○	○	○	●
Lego	○	○	●	○	●	○	○	●	○	○	○	●	●	○	●
Lichen	●	○	○	●	●	●	○	●	●	○	○	○	○	○	●
Pinktile	○	○	○	○	○	○	●	○	○	○	○	○	○	○	○
Proposte	○	●	●	○	●	○	○	●	○	○	○	●	●	○	○
Pulli	○	○	●	●	●	●	○	●	○	○	○	●	●	●	○
Sponge	○	●	○	○	●	○	○	●	○	○	○	●	●	○	○
Velvet	○	○	○	○	○	○	●	○	○	○	○	●	○	●	○
Walkway	○	●	○	●	●	○	○	●	○	●	●	○	○	○	●
Wallpaper	○	●	○	●	○	○	○	○	○	○	○	●	●	○	○
Wool	○	○	●	○	●	○	○	●	○	●	○	●	○	●	○

Table 3.1: BTFs with their tagged descriptors, according to our pilot study. Green ● and red ○ circles indicate whether the property has been associated or not to the BTF. The properties tagged are: (1) *high-contrast*, (2) *granular*, (3) *structured*, (4) *rough*, (5) *feature-dense*, (6) *complex-structure*, (7) *flat*, (8) *relief*, (9) *sharp-relief*, (10) *smooth-relief*, (11) *glossy*, (12) *color*, (13) *light*, (14) *soft*, and (15) *hard*.
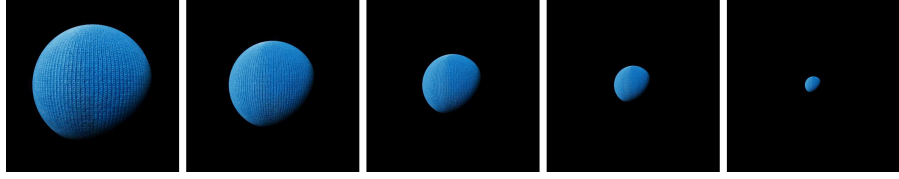
Figure 3.4: Sphere viewed under the different distances used in Experiment 1, from left to right d_0, d_1, d_2, d_3, and d_4. The sphere is rendered using BTF *Wool* iluminated with light direction l_1.

plest and most efficient; additionally, it is implemented by default in graphics hardware, although it has known poor performance with high frequencies [13]. The Gaussian filter usually behaves better, and offers a good compromise between final result and cost. Finally, the Lanczos filter is the finite filter that best models the ideal *sinc* kernel, but is the most computationally expensive. The implementation and parameters of the kernels follow PBRT [176]. We use isotropic filtering for the three cases. Anisotropic filtering would improve the results, so using isoptropic filtering is a *worst case* scenario. Additionally, it keeps the number of explored dimensions tractable.

PRE-FILTERING THE BTFS   If we opt for pre-filtering only in the spatial domain of the BTF, each texture image used to represent the BTF is mipmapped from a spatial resolution of 256×256 down to 1×1, thus yielding a hierarchy of nine levels. Each scene is rendered from five different distances d_i. The closest distance d_0 is chosen so that $r = 1$ (the ratio texel to pixel) in the center of the rendered image of the sphere. Successive distances
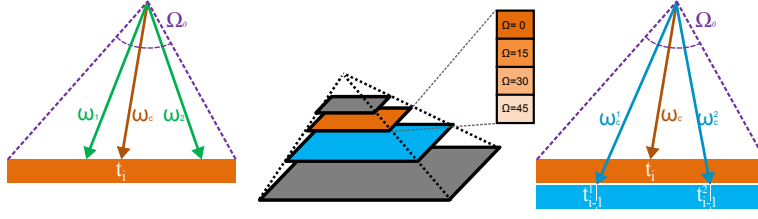
Figure 3.5: Left: Incoming light direction ω_i at each point in a texel varies from ω_c in the middle of the texel. Middle: To solve this, each level of the spatially filtered mipmapping hierarchy is extended to contain several filtered versions computed for different solid angles Ω. Right: Radiance in texel t for solid angle Ω_0 is obtained by sampling the immediate lower level texels in the hierarchy t_{i-1}^1 and t_{i-1}^2 at directions ω_c^1 and ω_c^2 respectively.

are set so that r at distance d_i is 2^2 times the ratio at d_{i-1}. Thus, for d_1 we have $r = 4$ and for the final d_4 we have $r = 256$ (Figure 3.4). Distances d_i vary directly with view vector v; in the following we use d_i as the distance where the objects are viewed, and v as the view vector.

Filtering only in the spatial domain in BTFs might lead to incorrect appearance, as described in Section 3.3 and shown in Figure 3.5 (left). Let Ω_0 be the effective solid angle subtended by a pixel and the light source, which accounts for the variation of the angle between the light direction and the normal in the points of the surface (see Figure 3.2). Assuming that all incoming light on a pixel is from the same direction ω_c (sampled at the center) is wrong, since different points receive light from different directions ω_i. Since these directions depend on Ω_0, we create several filtered versions for each level of the mipmap hierarchy, varying it from $0°$ to $45°$ in 15-degree increments (Figure 3.5 (middle)). We found that four levels are enough for our BTFs. Then, to accurately compute incoming light at a given filtered texel t_i, we take into account the immediate lower level in the hierarchy t_{i-1}, and sample its center point instead (Figure 3.5 (right)). The same approach works for the viewing direction v.

At rendering time, the pre-filtered hierarchy is accessed based on the ratio r for the spatial domain, which is computed using the derivatives of the texture coordinates at the pixel. This gives us a texel t at hierarchy level k. Then, the light and view solid angles are used to choose the pre-filtered version of t, as described above. Thus our multidimensional mipmapping hierarchy is accessed by specifying three input parameters: r, Ω_l (light) Ω_v (view). These last two are computed in run-time using the derivatives in the light and view directions with respect to the normal of the surface, which allows accounting for surfaces with a varying normal within a pixel. Note that in our experiments the normal varies smoothly within the pixel, so this variation can be captured using the pixel derivatives. For geometries with more complex geometry, level-of-detail techniques can be applied [142]. Linear interpolation between hierarchy levels is used.

Similar to the common bias term used in mipmapping, we introduce a scale factor s to guide the selection of the level in the pre-filtered hierarchy. This factor is used to scale the input parameters used to traverse the pre-filtered hierarchy r, Ω_l, and Ω_v. Intuitively, $s < 1$ means under-sampling the BTF, while $s > 1$ means blurring it. We note that two key characteristics that make BTF filtering interesting are the different nature of the spatial and angular domains, and the interplay between them. We thus define three different scales: s_x for the spatial domain, and s_l and s_v for the light and
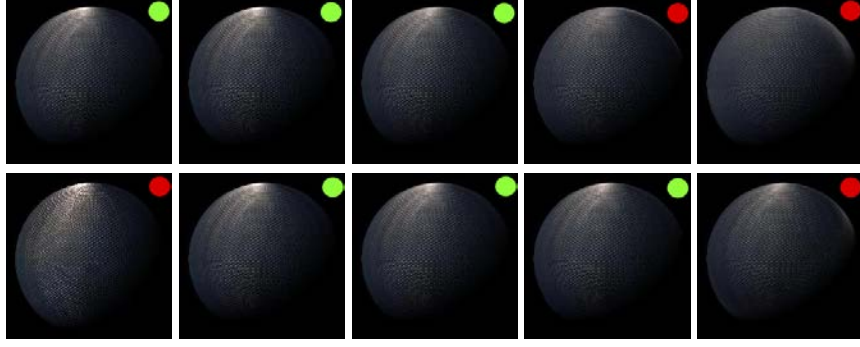
Figure 3.6: Example of filtering in the angular domain. The BTF *Cambrils* is filtered varying s_v (top) and s_l (bottom) with values (from left to right) .25, 1, 2, 8, and 32, at distance d_2 and illuminated from direction l_1. The inset dots means whether the image has been considered visually equivalent to a reference (green) or not (red), based on a pilot study.

the view directions in the angular domain, respectively. In our experiments, we set $s_x \in S_1 = \{.25, .5, 1, 2, 4\}$. For the angular domain, we set s_l and s_v to $S_2 = \{.25, .5, 1, 2, 4, 8, 16, 32\}$, based on an initial exploratory pilot study. Figure 3.6 shows an example of the *Cambrils* BTF under varying parameters of s_v and s_l. To be able to explicitly explore angular filtering while keeping the experiment tractable, we only vary one of the scale factors each time, while fixing the other two at 1 (no filter size biasing). Additionally, we also explore the interplay between domains by analyzing all five cases where $s_x = s_i = s_o = s$. This yields a total of 23 scale combinations.

COMPARISON WITH MIPMAPPING   A first concern is whether the proposed multidimensional filtering has advantages over classic, simple mipmapping in the spatial domain, which assumes that the normal, light and view directions remain constant within the pixel. We evaluate this by means of a pilot study, run on sixteen participants in our laboratory. The study followed a randomized two-alternative-choice (2AFC) design, similar to the used subsequently in Experiment 1 (Section 3.5). The user is shown test pairs consisting of an image filtered with either the multidimensional filtering or classic mipmapping, and a multi-sampled ground truth, and he has to select which of them represents a reference material more accurately. The results show that our multidimensional filtering performs significantly better ($F = 28.8$, $p < 0.01$) with a significance level of 99%, so it is indeed a better strategy for BTFs than mipmapping. Details about this pilot study can be found in Section 3.B.

MECHANICAL TURK   Similar to many previous studies in computer graphics [23, 191, 167, 11], we use Amazon Mechanical Turk (MTurk) as a source of participants for the experiments. A reasonable concern when using MTurk as a source of participants in user studies is the possible effect that uncontrolled viewing conditions (such as display resolution, brightness or environment light) may have in the data. On the other hand, using MTurk allows to have a significantly larger number of participants, much larger than what can be achieved in controlled lab sessions, which reduces variance. Heer and Bostok [77] showed that MTurk can actually be used in visual psychophysical experiments obtaining valid results; they replicated classic perception experiments in MTurk and compared the results with those from controlled

Figure 3.7: Sphere illuminated from the four light vectors l_i used in Experiment 1, from left to right l_1, l_2, l_3 and l_4. This example shows the sphere rendered using the BTF *Pulli* with scale $s = 1$ at distance d_2.
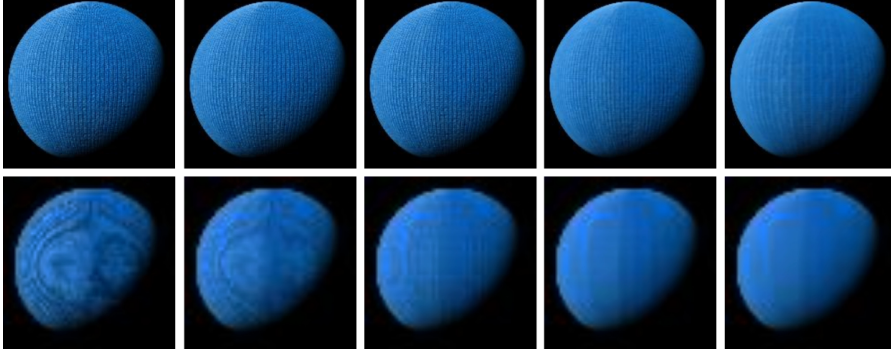
Figure 3.8: Example of filtering all domains with same scale s. The BTF *Wool* is shown at distances d_2 (top) and d_4 (bottom), filtered varying s with values (from left to right) .25, .5, 1, 2, and 4, illuminated from direction l_1.

experiments, finding a good match between the two sources of data. Nevertheless, we wanted to check the reliability of our MTurk data. We thus repeated the main experiments described in this paper under controlled conditions in our lab, and found that the results are indeed consistent. Details and analysis can be found in Section 3.D.

DATA ANALYSIS   We seek to establish a threshold where the visual equivalence between the reference and a given stimuli is lost, either due to aliasing or excessive blurring. Following previous work [183, 131], we cut through the data, using a conventional 75% 2AFC threshold value, where 50% is pure guessing. Additionally, to study the effect of each variable and to find significant trends in the data, we use N-ways Analysis of Variance (ANOVA), focusing on main effects and interactions between variables. Significant effects are further analyzed by using a Tukey-Kramer post-hoc analysis, to detect potential false-positives [24].

To obtain more meaningful analysis, we also analyze the effect of the high-level appearance descriptors explained in Section 3.4. In the following sections we describe each experiment, and summarize our main findings derived from them.

3.5   EXPERIMENT 1: STATIC SCENE

We first take into account both the spatial and angular domains on static images.

3.5.1   *Description*

STIMULI   The stimuli used are the result of combining our sixteen BTFs rendered at four different distances d using all 23 possible combination of scales $s =< s_x, s_i, s_o >$, for each of the three filtering kernels. To account for the light vector, we include four different incoming directions $l_i = \{\phi_i, \theta_i\}$, chosen to light the sphere from significantly different angles and offering a wide range of depictions. In particular, we take: $l_1 = \{-45°, 45°\}$, $l_2 = \{0°, 22°\}$, $l_3 = \{30°, 0°\}$ and $l_4 = \{100°, -28°\}$ (Figure 3.7). We define $\{\phi = 0, \theta = 0\}$ as the direction from the camera to the center of the sphere.

This makes a total of 17664 stimuli images. To handle such large number, we split the experiment into four smaller parts. In Part 1 we use the same scale for all domains ($s = s_x = s_i = s_o$), only one light direction (l_1), and the box filter, making a total of 320 test images. An example for BTF *Wool* and distances d_2 and d_4 is shown in Figure 3.8; the full set of stimuli can be found in the supplementary material. In Parts 2 and 3 we explore the light direction l and the filter kernel respectively, so we reduce the number of images in other dimensions. First, we reduce the number of BTF by clustering the original sixteen according to the results of the Part 1 (Section 3.5.2), and taking representatives from each cluster. This yields eight BTFs: *Cambrils*, *Corduroy*, *Impala*, *Proposte*, *Pulli*, *Velvet*, *Wallpaper* and *Wool* (please refer to Section 3.C for the full description of the clustering process). Additionally, we only use distances d_2 and d_4. Although no significant effect on d was found in Part 1 of the experiment, we opt for a somewhat conservative approach and analyze close/medium and long views. This makes a pool of 320 and 240 images respectively.

Finally, in Part 4 we explore the effect of scale s in the spatial and angular domains *independently*; as argued before, this is of particular interest for BTF filtering. We vary the scale factor in one domain at a time, while keeping the other two set to 1 (no scaling) (see Section 3.4). We again use the reduced stimuli from Parts 2 and 3 (eight BTFs, two distances d), using only the box filter. Since no significant effect with light direction l was found in Part 2, we only use two light directions l_1 and l_3. This gives us a total of 608 stimuli.

EXPERIMENTAL PROCEDURE   A total of 350 subjects took part in the first part of the experiment (236 male, 78 female; some participants did not report gender), 650 in the second (336 M, 233 F), 250 in the third (90 M, 140 F), and 625 in the fourth (300 M, 234 F), with ages between 18 and 64. All of them reported normal acuity and color vision; some of them had a computer graphics and/or artistic background. The minimum required screen resolution was 1024 × 768 pixels. A written explanation was given at the beginning of each experiment, although none of the participants were familiar with the final goal.

The experiment followed a fully randomized two-alternative-forced-choice (2AFC) design. Each participant was shown an average of around 20 tests, with a time limit of 20 seconds to answer each one. Three images are shown simultaneously, depicting a sphere with the same BTF. On the top, a multi-sampled close view of the sphere is shown as reference (rendered with super-sampling and jittered anti-aliasing to avoid artifacts, using a Gaussian filter for reconstruction); on the bottom, two test images from the stimulus set are shown. One shows the ground-truth, multi-sampled render of the BTF, while the other shows a pre-filtered representation, rendered with one sample-per-pixel. The position (left or right) is randomized. The two
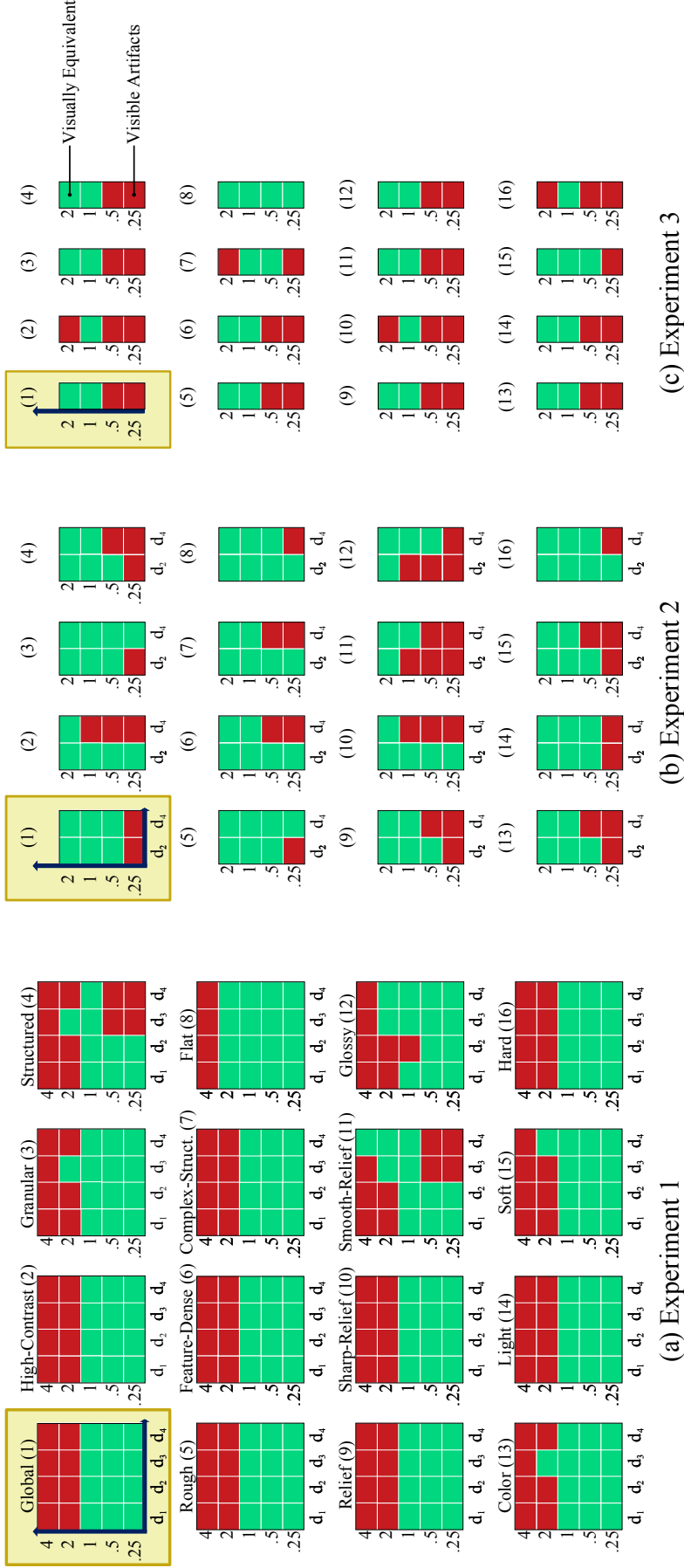
Figure 3.9: Results of (a) Experiment 1 (Section 3.5), (b) Experiment 2 (Section 3.6), and (c) Experiment 3 (Section 3.7), globally and for each property studied. Green means that visual equivalence is found, while red means that visible differences in form of artifacts are detected. For Experiment 1 and Experiment 2, each graph shows the effect on the visual equivalence for the combination of scales s (y-axis) and distance d (x-axis). For Experiment 3, visual equivalence depends only on the scales s (y-axis).
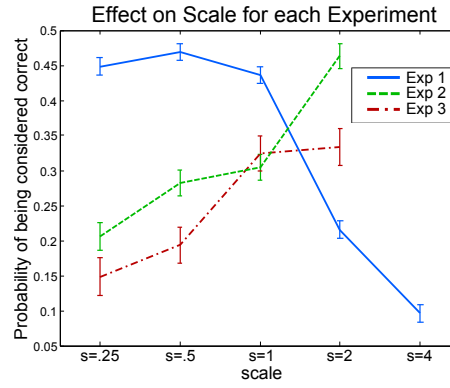
Figure 3.10: Main effects found for scale s in the results of Experiment 1 (Section 3.5), Experiment 2 (Section 3.6), and Experiment 3 (Section 3.7). In static scenes, visual equivalence drops significantly for over-blurred BTFs ($s > 1$). However, when introducing dynamics in the stimulus (i.e. moving light or camera), blur results into higher visual equivalence, in contrast with visual equivalence for under-sampled BTFs ($s < 1$). In the x-axis are the filter scales s, while the y-axis represents the visual equivalence for each scale.

images are rendered from a more distant point of view, and under a different illumination than the reference, to avoid matching tasks in image space. We apply an antialiasing mask to the edges of the spheres. The subject is asked: *"Which image represents the reference material on top more accurately?"*. We randomly introduced a few control tests with clear artifacts in one of the images, and discarded the results from participants who failed to provide the right answer to *all* of them, keeping in the end about 90 % of the total.

3.5.2   *Results*

Figure 3.9 (a) summarizes the results for the BTFs categorized with each property for the same scale for all dimensions; we further discuss each domain separately later in the section. Green indicates visual equivalence with the reference for a given distance d and scale s (75% 2AFC threshold), while red indicates visual differences in form of artifacts, globally and for each visual property used to categorize the BTFs. For the second and third parts of this first experiment, no significant differences were found between light directions l ($F = 1.34$, $p > 0.25$), nor between different kernels ($F = 1.32$, $p > 0.26$). A significant interaction effect between l and d was found ($F = 2.03$, $p < 0.05$), but the post-hoc test showed no significant differences between pairs. Finding no effect on the kernels used is surprising: one might expected that both the *Lanczos* and *Gaussian* kernels would performed significantly better than *box* filter. However, we found in this experiment that filter type did not affect the visual equivalence of BTFs. This is interesting when designing efficient rendering strategies.

A significant trend is found in scale s ($F = 198.05$, $p < 0.05$): under-sampled ($s < 1$) BTFs are overall considered visually equivalent to the reference, while over-blurred ($s > 1$) BTFs are, in most cases, found non-equivalent ($s = 4$ in particular is almost never considered visually equivalent). This indicates that aliasing on the surface appearance is preferred for static images. Blurring reduces contrast, and it is well-known that the human visual system is especially sensitive to contrast, which may explain this
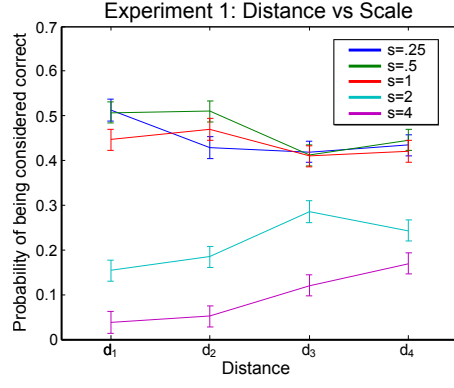
Figure 3.11: Interaction effect between *d* and *s* in Experiment 1. The trends shown that the even when *d* increases, under-sampled BTFs ($s < 1$) still perform better than over-blurring ($s > 1$) for our range of distances tested.
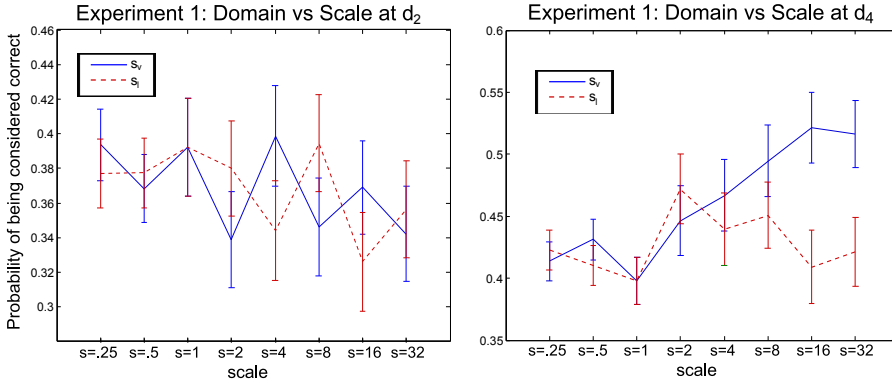
Figure 3.12: Effect of varying the scale on each of the two angular domains of the BTF (s_v, s_l) for distances d_2 (left) and d_4 (right) in static scenarios. Each domain is scaled independently, while fixing the other two domains to $s = 1$ (i.e. $s_i = 1$ is common for all domains).

result. A two-way interaction between filter scale *s* and distance *d* ($F = 5.22$, $p < 0.05$) shows that this trend is consistent across all distances tested (Figure 3.11).

Globally, no effect was found between distance *d* and visual equivalence ($F = 0.43$, $p > 0.73$). This is surprising, since it was expected that larger distances would be more forgiving. However, if we focus on individual properties, it can be seen that in *structured*, *repeatable* or *regular* BTFs when distance is increased, over-blurred BTFs are preferred over under-sampled, probably because artifacts in repeated structured patterns are easier to spot, in contrast with BTFs with *complex structure*, where aliasing is preferred.

BTFs with *relief* are significantly harder to filter than *flat* BTFs ($F = 34.56$, $p < 0.05$). This was intuitively expected, but our tests confirm the statistical significance of the difference. A closer look reveals another interesting finding: for *sharp* relief, under-sampling ($s < 1$) is always preferred, while blur ($s > 1$) is considered non-equivalent at any distance *d*; in contrast, for long distances, blur is preferred in BTFs with *smooth* relief.

We found a significant difference on sampling each domain separately: varying only the scale in the spatial domain s_x rates significantly lower than varying the scale of the angular dimensions ($F = 77.71$, $p < 0.05$), but do not present significant differences with scaling all domains at the same time
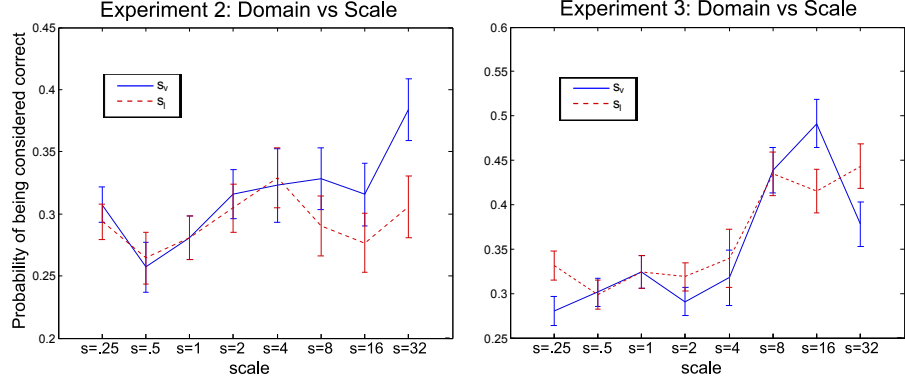
Figure 3.13: Effect of varying the scale on each of the two angular domains of the BTF (s_v, s_l) for moving the light source (left) and the camera (right). Only the scale of one domain is fixed at a time, while fixing the other two domains to $s = 1$ (i.e. $s_i = 1$ is common for all domains).

(i.e. $s_x = s_v = s_l$). This means that artifacts in the spatial domain due to under-sampling or blur are more noticeable than in the angular domain, which can be subsampled more aggressively. Focusing on the angular dimensions, the results show that errors in the light domain are easier to spot than in the view domain. A post-hoc test shows that this effect is dependent on the distance d to the object: for close-medium distance d_2 no significant trend is observed when varying s_v and s_l (Figure 3.12 (left)), whereas for longer views the view dimension can be further subsampled (Figure 3.12 (right)).

3.6 EXPERIMENT 2: TIME-VARYING ILLUMINATION

Experiment 2 analyzes the effect of dynamic illumination conditions, with the light vector l orbiting around the sphere.

3.6.1 *Description*

STIMULI   Adding a new parameter again makes the complete set of stimulus too large. Additionally, each test is now longer, since it involves animations. We thus perform additional simplifications in our multidimensional parameter space, based on the results of Experiment 1: we start from the reduced stimuli from Part 2 and Part 3 of Experiment 1 i.e. eight BTFs times two distances d. Since no significant effect on the filtering kernel was found (see Section 3.5.2), we only use the box filter, because it represents the theoretical worst-case scenario. We analyze the 23 different combinations of $< s_x, s_l, s_v >$, making a total of 368 animations.

EXPERIMENTAL PROCEDURE   A total of 517 (320 M, 160 F) different participants took part, with ages between 18 and 64. The experiment was carried out under similar conditions as Experiment 1. Training tests were presented prior to the experiment. Each test shows two animations (ground-truth and filtered) of the light vector l rotating around the sphere. No reference animation was shown, since three videos at the same time would be too confusing. The participants are given 30 seconds to answer each test. The video is played in loop mode until the user answers or time-out is reached. We ask: *"Which scene better represents the material without artifacts?"*. A brief explanation of what we mean by *artifacts* is given at the beginning. Since we

rely on a high-quality video reproduction, we add an additional confidence test: if a participant reaches time-out more than 3 times, we conservatively assume poor playback quality and reject the participant's data. In the end we kept the responses from about 80% of the participants.

3.6.2 *Results*

As before, we first focus on scaling all dimensions at the same time, and then we discuss each domain separately. Our results show that visual equivalence can be achieved in at least one pre-filtered representation for all BTFs (Figure 3.9 (b)). A significant effect was found for scale s ($F = 34.39$, $p < 0.05$, see Figure 3.10): Larger scales obtain better rates, which means that artifacts produced by undersampling ($s < 1$) the light domain become too noticeable. Even at close distances, over-blurred BTFs are considered visually equivalent. This is the contrary to the static case analyzed in Experiment 1, where contrast was the main feature to be preserved.

However, analyzing the individual properties of the BTFs, some other interesting properties can be inferred from our results: On most high-frequency BTFs where oversampling may blur out recognizable patterns (i.e. properties *high-contrast*, *complex-structure*, *sharp-relief* and *feature-dense*), some spatial aliasing is preferred at close distances.

Analyzing filtering each BTF domain separately, we find that, again, the angular domains can be filtered with values of s_v and s_l higher than the spatial scale s_x, confirming that artifacts in the spatial domain are more easily spotted by a human observer than in the angular domain. We found no significant effect between the different angular dimensions ($F = 2.66$, $p > 0.1$), nor in the interaction effect between the dimensions and the scale ($F = 0.17$, $p > 0.77$). However, an interesting trend can be observed for s_v (Figure 3.13 (left)), where over blurring in the view domain $s_v = 32$ rates significantly higher than the baseline $s_v = 1$.

3.7 EXPERIMENT 3: TIME-VARYING VIEW

Last, we study how the perception of pre-filtered BTFs is affected by changes in the view vector v.

3.7.1 *Description*

STIMULI   We use the eight BTFs selected previously, rendered using the scales s_i from Experiment 2 and the *box* kernel for filtering, and render an animation of a receding sphere, from d_0 to d_4. Since in Experiment 1 no significant effect was found on light direction l, we reduce the directions used to two, l_1 and l_3. This makes again a total of 368 animations.

EXPERIMENTAL PROCEDURE   A total of 545 (346 M, 168 F) participants took part of the experiment, with ages between 18 and 67. The videos are shown in loop mode, with the camera moving back and forth perpendicular to the screen, so that the projected area of the sphere decreases linearly. Again, no ground-truth animation is shown. The rest of the experiment (procedure and question asked) is similar to Experiment 2. We kept 75 % of the participants in this experiment.

3.7.2 *Results*

Figure 3.9 (c) summarizes our results for $s = s_x = s_v = s_l$. A clear pattern emerges: undersampled pre-filtered representations ($s < 1$) are not considered visually equivalent to the reference. This indicates that temporal aliasing artifacts due to varying the view vector v are the most disturbing. This holds true except for surfaces with very high-frequency appearance (*high-contrast*, *complex-structure* or *sharp-relief*), which confirms our findings in our previous experiment: The detailed appearance of these BTFs is over-smoothed, and the exact filter size should be used.

Comparing these results with Experiment 1, the visual equivalence of pre-filtered BTFs drops significantly (Figure 3.10). Under-sampled representations, which were considered equivalent in the static scenario, are now viewed as non-equivalent for most tested BTFs ($F = 34.29$, $p < 0.05$).

Again, we observe no significant differences between scaling all parameters and scaling only in the spatial domain s_x ($F = 1.33$, $p > 0.26$). However, there is an interesting trend as s_v and s_l increase, where the visual equivalence of the pre-filtered BTF also tends to increase (Figure 3.13 (right)). This again suggests that over-blurring the angular domain of the BTF is preferred by the human observer.

3.8 GENERALIZATION

We now show how our results generalize to more complex geometry and illumination configurations. Additionally, we discuss an automatic method to assign properties to the BTFs, based on statistical analyses.

GEOMETRY AND ILLUMINATION   We have conducted an additional study with new geometries and illumination conditions. We follow the work of Ramanarayanan et al. [183], and add two new geometries of increasing complexity: the Stanford Bunny and Dragon. For illumination, we introduce natural illumination modeled with the *Uffizi* and *Grace* environment maps from [28]. We choose these two maps as representatives of low- and high-frequency illumination, respectively. The generalization study was performed using the same procedure as Experiment 1, with two distances (d_2 and d_4) and the same scale s for the three domains. We compare the ground truth with multiple samples per pixel against a prefiltered version, with just one sample per pixel. From the intersection point of that sample 2048 rays are thrown over the hemisphere to compute the illumination from the environment map. We found no significant differences from using different geometry ($F = 0.97$, $p > 0.37$) or different illumination ($F = 1.54$, $p > 0.21$), so we conclude that our findings generalize well. Details and example stimuli for the BTF *Cambrils* can be found in the Section 3.E and Figure 3.20 respectively. Figure 3.14 illustrates this generalization: the BTFs shown are rendered using filtering scales $s \neq 1$ for distances d_2 and d_4, which were found to be visually equivalent in Experiment 1 (see Section 3.5). Figure 3.15, produced with the proposed rendering algorithm explained later in this section, shows additional evidence.

STATISTICAL ANALYSIS   In this work we have used high-level descriptors of the BTFs. Adding such tags would be a simple, non-taxing step in future acquisition protocols. However, for existing untagged BTF databases, automatic classification is also interesting. As a first step towards such automatic

Figure 3.14: Examples of equivalent pre-filtered representations of BTFs for different geometries under natural illumination. The larger image shows the multi-sampled reference image at close distance, while the smaller images show the visually equivalent, pre-filtered representations for distances d_2 and d_4, and scale $s = .25$. From left to right: *Wallpaper*, *Cambrils* and *Pulli*.

model, we have explored how a set of low-level statistics correlate with our categorization and our results. These statistics include common first-order image statistics [177] and textural features [209] on both the spatial and the angular domains, plus the BTF statistics proposed by Filip et al. [39]. We have measured the correlation between these statistics and the results from the pilot categorization study using Pearson correlation [24], which evaluates the linear association r between the two variables tested, with $r \in [-1, 1]$ ($r = 1$ being perfect linear correlation, $r = -1$ perfect inverse correlation, and $r = 0$ no correlation). We only keep correlations with a significance of 95% ($p < 0.05$). We have found several strong correlations ($|r| > .8$) for some properties. For instance, between *Granularity* and the directionality in the angular domain (kurtosis and skewness of the histogram of directions), between *Glossiness* and the kurtosis and skewness of luminances, or between *Structureness* and the directionality statistics in the spatial domain. Please refer to Section 3.F for the complete description of the statistics analyzed, and the full list of correlations found. While finding a complete and robust set statistics that match higher-level perceptual properties of BTFs is outside the scope of this paper, we believe these preliminary findings show great promise, while opening a new avenue of interesting future work.
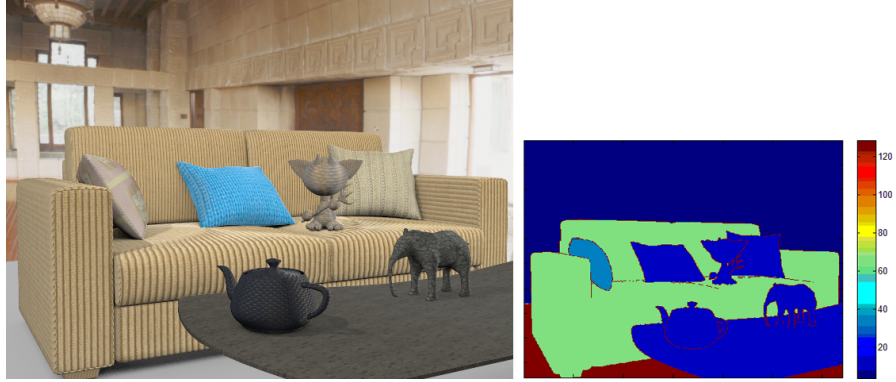
Figure 3.15: Example of the application on adaptive rendering using our findings to guide sampling. For each ratio *texel* : *pixel* defined by the projection of the pixel in the BTF, the visual properties of the BTFs are used to guide the number of samples (left) on the surface, based on the results summarized in Figure 3.9. The base samples-per-pixel spp_0 is 128. Note that the sampling is guided only by the visual properties of the surface; surfaces with no tagged descriptors (e.g. the floor) receive spp_0 samples-per-pixel (spp). Using this approach we use an average of 51.2 spp, as opposed to the 128 spp needed without using any adaptive scheme.

3.9   PRACTICAL APPLICABILITY

We propose a set of practical applications to improve the performance of rendering and compression of BTFs without compromising their final appearance.

FILTERING KERNELS   The first immediate application of our study has to do with the choice of filtering kernel: No significant effect was found from using different kernels in static scenes. This means that the cheapest filter kernel (box filter) can be used in several scenarios. In our implementation, this is more than 8 times faster than Gaussian and Lanczos filters.

Figure	PF Time	MS Time	Speed-Up
Figure 3.14, *Wallpaper*	6′47″	56′32″	×8.33
Figure 3.14, *Cambrils*	7′26″	1h05′01″	×8.74
Figure 3.14, *Pulli*	2′24″	16′15″	×6.77
Figure 3.16, *Impala*	6′47″	56′57′	×8.39
Figure 3.16, *Wool*	0′1.8″	1′10″	×61.1

Table 3.2: Rendering times for the images in Figure 3.14 and Figure 3.16, both using the pre-filtered representation (PF) and the multi-sampled ground truth (MS). Image *Wool* is illuminated with one light sample per pixel; this makes its speed up increase significantly.

RENDERING   For all properties considered in all different scenarios tested (static, moving light and moving camera), there is at least one pre-filtered representation of the BTF (rendered with just one sample) that is visually equivalent to a multi-sampled reference (see Figure 3.9). This has direct applications in rendering, since it means that the isotropic pre-filtering proposed in Section 3.4 can be used with just one sample per pixel, achieving visual equivalence to a multi-sampled reference. Figure 3.16 shows side-by-side comparisons between our pre-filtered versions and the multi-sampled ground truth, rendered using ray-tracing. While not identical pixel-wise, our pre-filtered versions are considered visually equivalent, and have been rendered with a speed-up factor of up to 61.1× in the most extreme case, and between 6× and 9× on average (see Table 3.2). Additional results and comparisons, including timings for each of them, can be found in the supplementary material[2].

Another rendering application lies in the context of adaptive rendering schemes, based on the visual properties of BTFs. It works as follows: once the ray hits a surface shaded with a BTF, we can access the visual properties defining the underlying BTF (manually tagged, or explored through statistical analysis as explained above). With these properties, and the value d obtained using the texel footprint in the pixel (computed with the derivatives of the ray in the pixel), we obtain the minimum scale s that achieves visual equivalence (Figure 3.9). This scale s determines the number of samples per pixel spp_p needed to get good results at pixel p ($spp_p = s * spp_0$, with spp_0 a predefined, baseline samples-per-pixel value). Following the results from Experiment 1 (Section 3.5), we can reduce the number of samples up to a factor of 16 in static images. Figure 3.15 shows an example of a scene rendered using this adaptive approach. Note that this scheme only adapts the sampling to the surface being rendered, taking into account the surface properties. Other adaptive sampling schemes, such as sampling complex illumination patterns, are not handled by this scheme.

Our third rendering application involves BTF level-of-detail techniques such as the work by Ma et al. [146]. The authors build a Laplacian pyramid to model the pre-filtered LoD hierarchy. This is traversed for reconstruction during rendering time, starting from the lowest frequency (most pre-filtered) and adding higher-frequencies until reaching the appropriate level of detail. Based on the results summarized in Figure 3.9, the traversal of the pyramid can be stopped at lower-frequency levels, reducing reconstruction costs while maintaining visual equivalence for $s > 1$.

---

2 http://giga.cps.unizar.es/~ajarabo/pubs/btfTVCG14/index.html
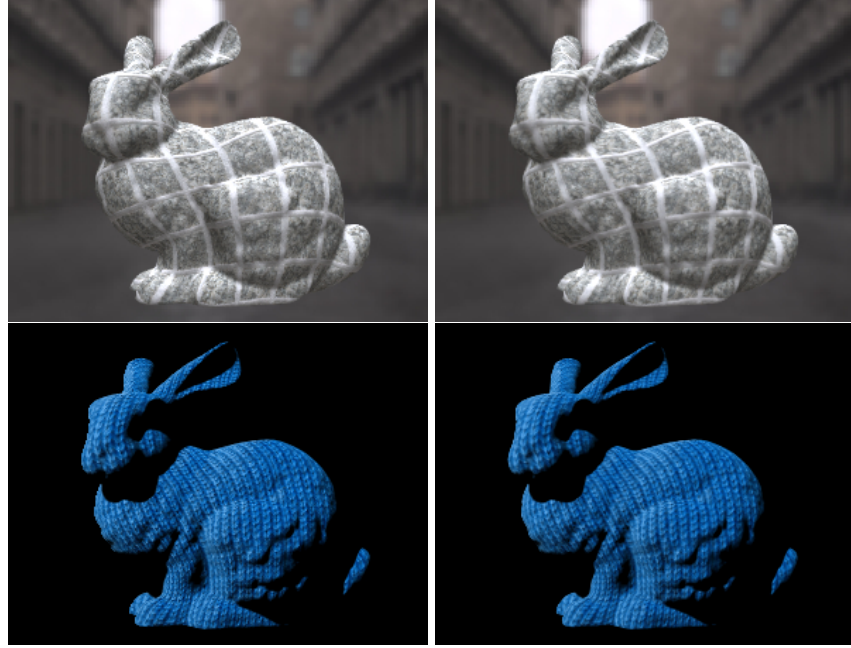
Figure 3.16: Comparison between a pre-filtered BTF (left) and its multi-sampled ground truth (right) at distance d_2. Top: *Impala*. Bottom: *Wool*.

COMPRESSION   Over-blurred BTFs are in general considered visually equivalent for dynamic scenes. This has two immediate consequences in compression schemes: first, it allows us to discard high-resolution levels in the level-of-detail hierarchy, which effectively reduces memory requirements. Second, since low-frequencies generally contain most of the signal energy while being easier to approximate without noticeable artifacts, more aggressive compressions can be employed. As a proof-of-concept, we have implemented a common PCA-based BTF compression algorithm [195], and applied it to all our BTFs using $s = 2$ and $s = 4$. On average, using the same reconstruction error threshold, we keep 99% of the original signal energy using only 28.5% and 8% of the coefficients needed for the original version, respectively (Figure 3.17).

Finally, the fact that the angular domain may be aggressively filtered without decreasing visual equivalence can also be applied to compression. Blurring in the angular domain makes it possible to represent angular effects with a small number of spherical harmonic basis functions. Representation with a small number of spherical harmonic terms is also advantageous for rendering with Precomputed Radiance Transfer techniques [204].

3.10 DISCUSSION AND FUTURE WORK

Important conclusions can be drawn from our experiments. For static scenes, there is a clear preference for under-sampled images, which might be due to the fact that humans are very sensitive to contrast, which decreases with blurring. The trend is reversed when considering motion (both light and camera). This may be due not only to the lack of flickering artifacts, but also to motion de-blurring mechanisms in our visual system, which make moving images appear sharper than they actually are [14]. This preference seems to be related to the motion blur effect [163]. Future work needs to
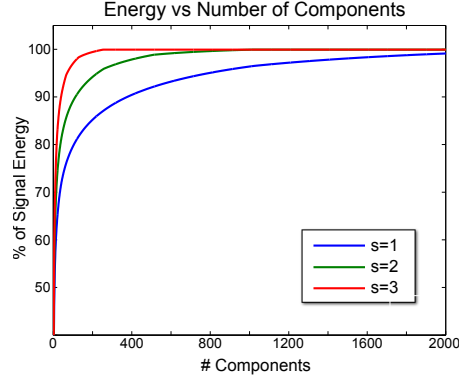
Figure 3.17: Percentage of the signal energy stored by the N principal components (x-axis) of the compressed BTF, averaged for all BTFs. Each line represents the results of compressing the original BTF ($s = 1$, blue), and its overblurred versions with $s = 2$ (green) and $s = 4$ (red). We refer to the supplementary material[3] for the individual graphs of each BTF.

be conducted to examine the relationship between pre-filtering and motion blur.

We have shown that filtering in the spatial domain rapidly affects visual equivalence, while the angular domain can be more aggressively filtered. In fact, the trend observed in the angular domain is that over-filtering is preferred over under-sampling; this supports the results from our pilot study, which shown that a multidimensional filtering produces more visually equivalent results than just filtering in the spatial domain (i.e. $s_v = s_l = 0$). In particular, at long distances the domain of the viewing vector can be filtered the most before losing visual equivalence, specially at long distances.

Another important finding is that high-level properties of the BTFs have a significant effect on how filtering affects appearance perception. For instance, under-sampling BTFs with clear symmetric structure quickly introduces artifacts that affect visual equivalence. This generalizes a conclusion of a recent study on media retargeting [191], which also identified structure as one of the most important image features that should be preserved. On the other hand, BTFs with complex high-frequency structure are in general more forgiving with under-sampling, specially for static scenes. This is in accordance with previous work [38], possibly due to the effect of visual masking (high-frequency features mask high-frequency artifacts). For BTFs with sharp relief, under-sampling $s < 1$ is preferred, even at the cost of visible aliasing. This is probably because relief is a very dominant and salient feature with strong parallax effects, which give the BTF a very three-dimensional, appealing appearance. In static BTFs with a high specular component, under-sampling is in general preferred. This makes sense since the specular highlight is usually a salient high-frequency feature. On the other hand, on animated scenarios, this behaviour is again reversed: aliasing becomes too distracting, making over-blur preferred by the observers. These findings suggest that high-level descriptors of BTFs could be used when devising an optimal sampling strategy, as we have shown in Section 3.9. We have found some correlation between these high-level BTF descriptors and low-level statistics, which we hope spur future work in this direction. Last, we have proposed several applications in BTF filtering, compression and rendering, directly derived from our results.

Although we have shown that our results generalize well, our work is limited by our choice of stimuli and filtering parameters. More BTFs could be added to future experiments, along with different lighting schemes, parameters, BTF descriptors and the combinations of these. Additionally, our conclusions are restricted to materials that can be accurately represented using BTFs, which are in general diffuse or moderately glossy. Although our work focuses on BTFs, we believe that some of the insights and methodology could be extrapolated to study the perception of other reflectance representations, such as SV-BRDFs, provided that the filtered surface's reflectance has moderate angular frequency. We hope that new studies and potential future applications can leverage our stimuli set, which can be accessed at the project page[4].

APPENDICES

3.A   CATEGORIZATION OF THE BTFS

The BTFs used in our experiments represent a wide range of materials, which makes the experiments hard to analyze globally. To obtain more meaningful data from the experiments, we categorize them into a set of high-level visual properties, by means of a pilot study. We build our set of categories from previous works on texture [184] and BRDF [151] naming. The categories used in the study, together with the textual description given to the participants, are:

HIGH-CONTRAST The surface presents, or no, high contrast in its features

GRANULAR Is the texture granular (i.e. it presents small micro-scale structure)?

STRUCTURED Does the surface present a clear structure or is it just random?

ROUGH Is the surface rough or smooth?

FEATURE-DENSE Does the surface presents several visual features (small details) or is mainly plain (even having some isolated detail)?

COMPLEX-STRUCTURE Has it a complex structure?

FLAT Is it flat...?

RELIEF ...Or does it present relief?

SHARP-RELIEF If the texture presents relief, does it present sharp edges...?

SMOOTH-RELIEF ...Or more like smooth bumps?

GLOSSY Does it have glossy (i.e. specular) appearance?

COLOR Is the surface colored or gray-scale?

LIGHT Is the surface albedo light or dark?

SOFT Does the surface has smooth appearance?

HARD Does the surface has hard appearance?

---

4 http://giga.cps.unizar.es/~ajarabo/pubs/btfTVCG14/

Prior to the experiment, a short explanation of each category was given to the ten participants. They were shown two static images of the same BTF, rendered on a plane from different points of view and illuminated from different directions (see Figure 3.18), and they had to indicate if the displayed BTF showed or not each individual property. No tiling was applied to the BTF to avoid introducing bias.

To determine whether a BTF presents a category, we cut through the data, using a conventional 75% threshold value, where 50% is pure guessing. The final assignment of categories with BTFs can be found in Table 3.1. The results have a confidence interval ranging from $\pm 0\%$ to $\pm 39\%$. Note that these confidence intervals are quite big, since the number of participants in this pilot study was small.
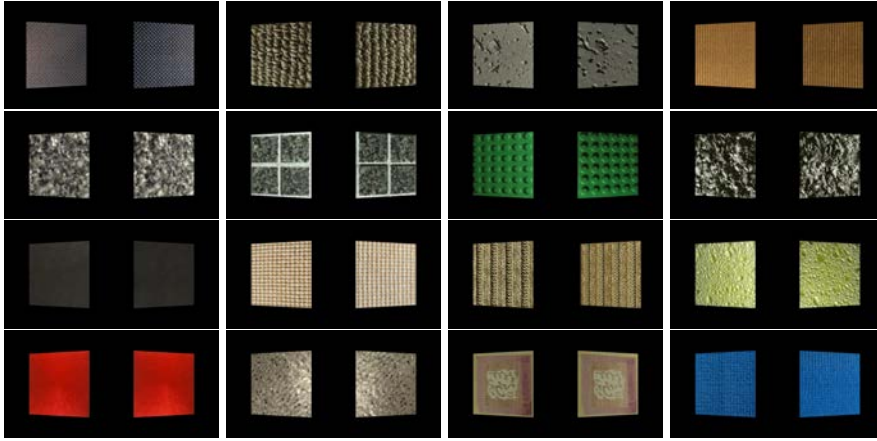
Figure 3.18: Images shown to the participants of the categorization experiment, for each BTF. In reading order: *Cambrils*, *Carpet*, *Ceiling*, *Corduroy*; *Floortile*, *Impala*, *Lego*, *Lichen*; *Pinktile*, *Proposte*, *Pulli*, *Sponge*; *Velvet*, *Walkway*, *Wallpaper*, and *Wool*.

3.B   PILOT STUDY: COMPARISON AGAINST MIPMAPPING

In this pilot study we explore the differences in terms of visual equivalence when pre-filtering a BTF using classic mipmapping [244] (filtering only in the spatial domain), and using the multidimensional pre-filtering described in Section 3.4.

STIMULI   The stimuli used in this pilot experiment is the result of combining the sixteen BTFs used in this work, rendered at 4 different distances d using five scales s, rendered using mipmapping or our multidimensional filtering, making a total of 640 images. The box filter is used to pre-filter the BTFs, and we limit the light direction to l_1, in order to keep the experiment tractable.

EXPERIMENTAL PROCEDURE   Sixteen subjects took part in the experiment (12 male, 4 female), with ages between 23 and 32 years old. All of them had normal acuity and color vision; some of them had a computer graphics and/or artistic background. A brief explanation about the test was given orally and in writing at the beginning, although none were familiar with the final goal of the experiment. All tests were run on a LCD screen set to its factory settings, with standard office lighting. The experiment followed
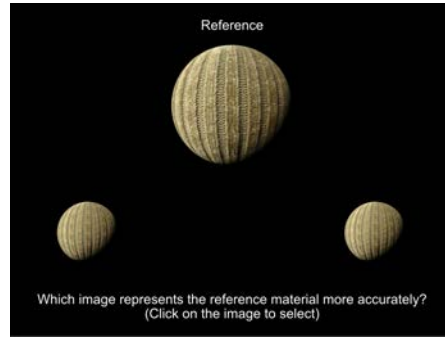
Figure 3.19: Interface used in the pilot study that compares between mipmapping and the proposed multidimensional filtering.

a fully randomized two-alternative-forced-choice (2AFC) design. There was no time limit in the experiment, which took an average of 17 minutes to complete.

For each test, three images were shown simultaneously, all of them depicting a sphere with the same BTF and lighting conditions (see Figure 3.19). On the top, a multi-sampled close view version of the sphere is shown as reference (rendered with super-sampling and jittered anti-aliasing to avoid artifacts); on the bottom, two test images from the stimuli set are shown. These are rendered from a more distant point of view, to avoid matching tasks in image space from the participants. One image shows the ground-truth, multi-sampled render of the BTF, while the other shows a pre-filtered representation (mipmapping or our multidimensional filtering) . The position of these two images (left or right) is randomized. The subject is asked the following question: *"Which image represents the reference material on top more accurately?"*.

RESULTS   Our analysis shows that using our multidimensional filtering performs significantly better than pre-filtering only in the spatial domain with classic mipmapping ($F = 28.8$, $p < 0.05$), with a strong interaction effect for s ($F = 165.8$, $p < 0.05$). Additionally, other interaction effects are found for several visual properties. The most interesting is the significant effect that appears in BTFs with *relief*; these are considered visually equivalent significantly more often when pre-filtered with multidimensional filtering than when using mipmapping ($F = 3.16, p < 0.05$). This suggests that for BTFs with visible self-shadows and parallax, it is necessary to filter all dimensions in the BTF; this is one of the most notable properties introduced by BTFs, as opposed to regular two-dimensional textures.

3.C CLUSTERING OF BTFS

Here we describe how to select a representative subset from the sixteen original BTFs used in Part 1 from Experiment 1 (Section 3.5).

We first removed *Walkway*, since we found visible artifacts when rotating the light due to discontinuities in the original BTF data; this would lead to problems in Experiment 2, orthogonal to the purpose of the study. To select between the remaining fifteen BTFs, we cluster them hierachically in a binary tree. This clustering groups the BTFs according to their Euclidean distance in a 20-dimensional space, defined by the results obtained combining five distances d and four scales s in Experiment 1.

Clustering is performed using the Matlab functions *linkage* and *cluster*. Then, we take the clusters with smaller distances between their elements, and remove one of them from the selection. Figure 3.20 shows the result of the hierarchical clustering performed. In the end, we keep eight BTFs: *Cambrils*, *Corduroy*, *Impala*, *Proposte*, *Pulli*, *Velvet*, *Wallpaper* and *Wool*.
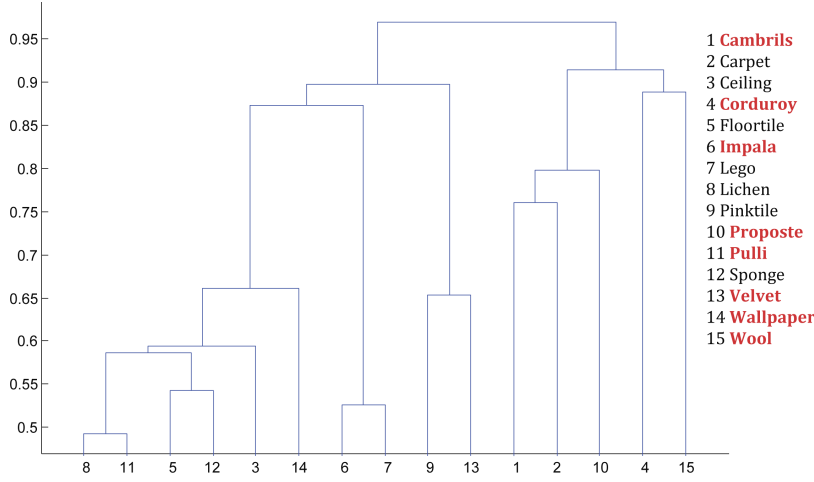
Figure 3.20: Hierarchical clustering of the BTFs based on the similarity of their results in Experiment 1 (Part 1). This clustering is used to reduce the amount of BTFs in the subsequent experiments. The BTFs kept for the experiments are highlighted in red.

3.D COMPARISON BETWEEN CONTROLLED AND UN-CONTROLLED EXPERIMENTS

Here we evaluate the potential effect that using MTurk may have on the data; to do so, we partially repeat the experiments described in the main text, under controlled lab conditions. In particular, we repeat Part 1 from Experiment 1 (Section Section 3.5), and Experiments 2 (Section Section 3.6) and 3 (Section 3.7). We refer to their particular sections in the main text for the full description of these experiments and the stimuli used.

EXPERIMENTAL PROCEDURE   A total of sixteen subjects took part in the in-situ Experiment 1 (12 male, 4 female), with ages between 23 and 32 years old. All of them had normal acuity and color vision; some of them had a computer graphics and/or artistic background. A brief explanation about the test was given orally and in writing at the beginning, although none were familiar with the final goal of the experiment. All tests were run on a 21" BENQ GL2240 LCD screen set to its factory settings, with standard office lighting. Each participant was shown the full stimuli in random order, making a total of 320 tests. It took an average of 17 minutes to complete the experiment.

Another sixteen different participants took part in the in-situ Experiments 2 and 3, equally divided between male and female, with ages between 23 and 41. The experiment was carried out under similar conditions as Experiment 1 (display, lighting, no time limit). Each participant answered all tests (64 tests in Experiment 2, 32 tests in Experiment 3), taking an average of 10 and 5 minutes to complete each experiment, respectively.
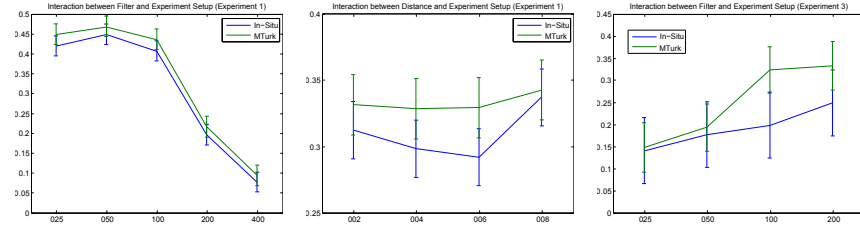
Figure 3.21: Results from the comparison between the experiments performed under controlled (in-situ) and uncontrolled (MTurk) conditions. From left to right, results from Experiment 1 (static) for scale *s*, and for distance *d*, and from Experiment 3 (dynamic view) for scale *s*. The results show no significant differences between factors, and that the trends are consistent in both scenarios.

RESULTS   We compare the results from the in-situ experiment, performed under controlled setup, with those obtained using MTurk. We seek significant differences between the source of the data, by using N-ways Analysis of Variance (ANOVA), focusing on both main and interaction effects with significance of 95%. Significant effects are further analyzed by using a Tukey-Kramer (or Tukey Honesty Significant Differences) post-hoc analysis [24]; this test is a modified t-test, that compares multiple means, assuming the null hypothesis that all means are equal. This test is appropriate for detecting false-positives that might be found by ANOVA, specially in multiple-factors experiments as ours.

On the static configuration studied in Experiment 1 we found a significant effect, showing a decrease in the performance of filtering for the in-situ experiments ($F = 8.76, p < 0.05$). However, the post-hoc test reveals that there is no significant differences between pairs in any of the dimensions analyzed: we found no significant differences on the results for BTF, distance *d* or scale *s* between the in-situ and the MTurk experiments (Figure 3.21 (left and middle)).

The results from Experiment 2, where we analyze the effect of moving light source, shown no significant effect on using MTurk in comparison with the data obtained in the controlled scenario ($F = 0.12, p > 0.72$).

Finally, the results of the comparison of the data in Experiment 3 (moving camera) shows that visual equivalence was found significantly more often in MTurk than in the in-situ experiment ($F = 7.34, p < 0.05$). Again, we analyze further the results using a post-hoc test: it shows that no significant differences between pairs appeared in the dimensions explored (Figure 3.21 (right)), even this main effect is found.

3.E   GENERALIZATION EXPERIMENT

We evaluate here how well our results generalize under less restrictive setups, introducing new geometries and illumination (see Figure 3.22).

GEOMETRY   We test images with three levels of increasing geometric complexity (*sphere*, *bunny*, *dragon*), which has been found to have a significant effect on material appearance [183, 131].

ILLUMINATION   Similarly, we test three levels of increasing illumination frequency: The *Uffizi* and *Grace* environment maps [28], plus a directional light.
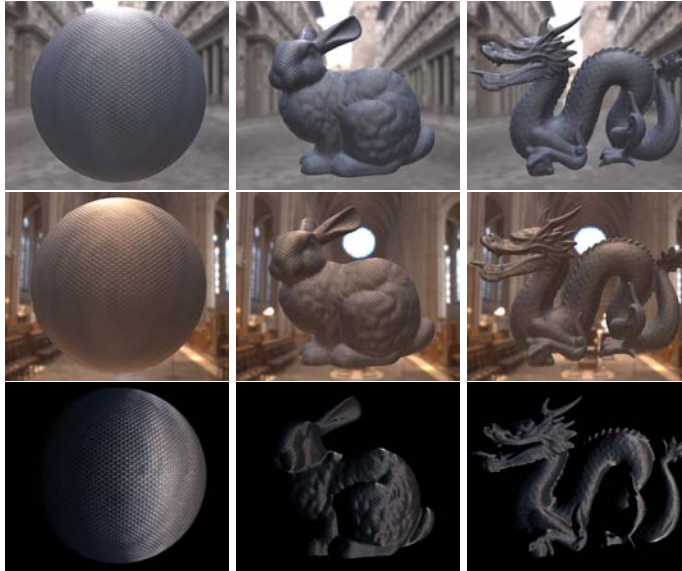
Figure 3.22: Reference renderings, using the BTF *Cambrils*, of the scenes used in the validation experiment. From left to right, the three geometries used: *sphere*, *bunny*, and *dragon*, in order of geometrical complexity according to Ramanarayanan et al. [183]. Top to bottom, the illumination setups used, ordered from low to high frequency: *Uffizi*, *Grace*, and *directional light*.

STIMULI AND PROCEDURE   We use the reduced stimuli from Part 2 in Experiment 1 (Section 3.5), with a *box* filter kernel. For direction illumination we use l_1. For the images illuminated by an environment map, up to 2048 cosine-weighted samples are traced. We blur the background during rendering to avoid masking effects in the appearance of the tested object. The test is carried out as in Experiment 1. A total of 256 subjects (131 M, 99 F) took part, from which we kept 95% for the analysis.

RESULTS   Both *bunny* and *dragon* perform slightly better than *sphere*. This is in accordance with previous work [183, 39, 131], which suggested that more complex objects are more forgiving to artifacts. In our particular case, however, the results show no significant effect ($F = 0.97, p > 0.37$). Similarly, no significant effect was found on illumination ($F = 1.54, p > 0.21$), with all studied light sources (*directional*, and *Grace* and *Uffizi* environment maps) performing similarly. Together, these results confirm that our findings do generalize to more complex geometry and lighting.

3.F   LOW-LEVEL STATISTICS OF BTFS

In order to give a first step on automatic extraction of high-level descriptors of BTFs, we investigate different low-level statistics of the BTF. The analysis covers a set of metrics based on first-order image statistics [177] and on a set of textural visual features [209]. This statistics have been computed for both different representations of the BTF: images in the spatial domain (i.e. photographs of the BTF taken under different light and view direction), and images in the angular domain (i.e. per-pixel BRDF representation).

Domain	General Stat.	Image Stat.	High-Level Prop	r
VL	Mean	LKurtosis	Granular	-0.812236
VL	Mean	DirKurtosis	Granular	-0.811917
VL	Mean	DirKurtosis	Flat	0.805539
VL	Median	DirSkewness	Granular	-0.829850
VL	Median	DirKurtosis	Granular	-0.877730
VL	Median	DirSkewness	Flat	0.810800
VL	Median	DirKurtosis	Flat	0.816263
VL	STD	LSkewness	Granular	-0.836658
VL	Skewness	LMean	Glossy	0.817957
VL	Kurtosis	LMean	Glossy	0.844585
ST	Mean	DirSkewness	Structured	-0.822910
ST	Mean	LSkewness	Glossy	0.803670
ST	Mean	LKurtosis	Glossy	0.822522
ST	Median	DirSkewness	Structured	-0.821881
ST	Median	LSkewness	Glossy	0.808776
ST	Median	LKurtosis	Glossy	0.828666
ST	Max	DirSkewness	Structured	-0.863716
ST	Max	DirKurtosis	Structured	-0.843974
ST	Max	DirSTD	Smooth	-0.806921

Table 3.3: Significant correlations found between low-level BTF statistics and high-level visual properties of the BTF. *Domain* is the space where the statistics have been computed (VL: view-light, ST: spatial); *General Stat.* is the metric used to integrate the measures for all images into a single value for each BTF; *Image Stat.* is the statistic computer, which can be a first-order statistic or a textural feature. The prefix "L" refers to luminances, while "Dir" refers to the directionality of the texture. Note that no value obtained with the measures of coarseness and contrast present significant correlation with visual properties.

For each image (in the spatial or angular domain) we compute several statistics: mean, standard deviation, skewness and kurtosis of the luminance channel of the image. Additionally, we compute the three most significatives textural features defined by Tamura et al. [209]: coarseness, contrast and directionality. We refer to the original paper for the mathematical description of these features.

Then, to compute the general metric of the full BTF, we compute the mean, median, standard deviation, skewness, kurtosis, maximum and minimum, in both the spatial and the angular domains. Note that the statistics used by Filip et al. [39] are included within this set of metrics.

To measure the correlation between the computed BTF statistics and the high-level descriptors presented in Section 3.A we have used Pearson correlation [24], which evaluates the linear association r between the two variables tested, with $r \in [-1, 1]$ ($r = 1$ being perfect linear correlation, $r = -1$ perfect inverse correlation, and $r = 0$ no correlation). We only keep correlations with a significance of 95% ($p < 0.05$). The results of the significant correlations are listed in Table 3.3.

Part III

TEMPORAL DOMAIN

This part is devoted to the *temporal domain* of the plenoptic function. In particular, we focus on time-resolved light transport, where the speed of light cannot be longer considered infinite. We start by describing the *femto-photography* technique, which is the first imaging system capable of imaging at a temporal resolution of picoseconds in macroscopic scenes, and the data processing needed for adequate visualization. We follow describing our framework for effectively simulating transient light transport, taking into account the particularities of the radiance in the temporal domain. Finally, we demonstrate a visualization system for time resolved data handling the relativistic effects due to ultra-fast camera motion.

FEMTO-PHOTOGRAPHY: CAPTURING AND VISUALIZING THE PROPAGATION OF LIGHT

In this chapter we present a system for imaging with an effective temporal resolution in the order of picoseconds, and the reconstruction and processing techniques needed to correctly visualize the captured data. media. This work originates as a colaboration between our lab and the Camera Culture Group at MIT Media Lab, inventors of the technique. My role in the project was fundamentally on devising the time-unwarping method Section 4.6 for correcting the temporal delays causing unintuitive effects as a post-process. We describe here the full project for completeness and context. This work was published in *ACM Transactions on Graphics* and presented at *SIGGRAPH 2013*, and was selected in the *ACM Best of 2013* list.

This work was the first on demonstrating time-resolved light transport in macroscopic scenes. Since the initial publication of this work numerous works [78, 111] have advanced this field by improving numerical models and introducing new, more accessible capture technology. In addition, numerous applications on scene understanding have been developed using this new type of imaging [223, 245, 79, 80]. Given the impact of the work, it was later invited to be published as a *Research Highlight* in the journal *Communications of the ACM*.

A. Velten, D. Wu, A. Jarabo, B. Masia, C. Barsi, C. Joshi, E. Lawson, M. Bawendi, D. Gutierrez & R. Raskar
Femto-Photography:
Capturing and Visualizing the Propagation of Light
ACM. Trans. Graph. Vol.32(4), SIGGRAPH 2013

A. Velten, D. Wu, B. Masia, A. Jarabo, C. Barsi, C. Joshi, E. Lawson, M. Bawendi, D. Gutierrez & R. Raskar
Transient Imaging of Macroscopic Scenes at Picosecond Resolution
Communications of the ACM 2015, *to appear*

4.1 INTRODUCTION

Forward and inverse analysis of light transport plays an important role in diverse fields, such as computer graphics, computer vision, and scientific imaging. Because conventional imaging hardware is slow compared to the speed of light, traditional computer graphics and computer vision algorithms typically analyze transport using low time-resolution photos. Consequently, any information that is encoded in the time delays of light propagation is lost. Whereas the joint design of novel optical hardware and smart computation, i.e, computational photography, has expanded the way we capture, analyze, and understand visual information, speed-of-light propagation has been largely unexplored. In this work, we present a novel ultrafast imaging technique, which we term *femto-photography*, consisting of femtosecond laser illumination, picosecond-accurate detectors, and mathematical reconstruction techniques, to allow us to visualize movies of light in motion as it travels through a scene, with an effective framerate of about

Captured Movie

Streak image

Peak-time visualizations

[ps] 200 600 1000
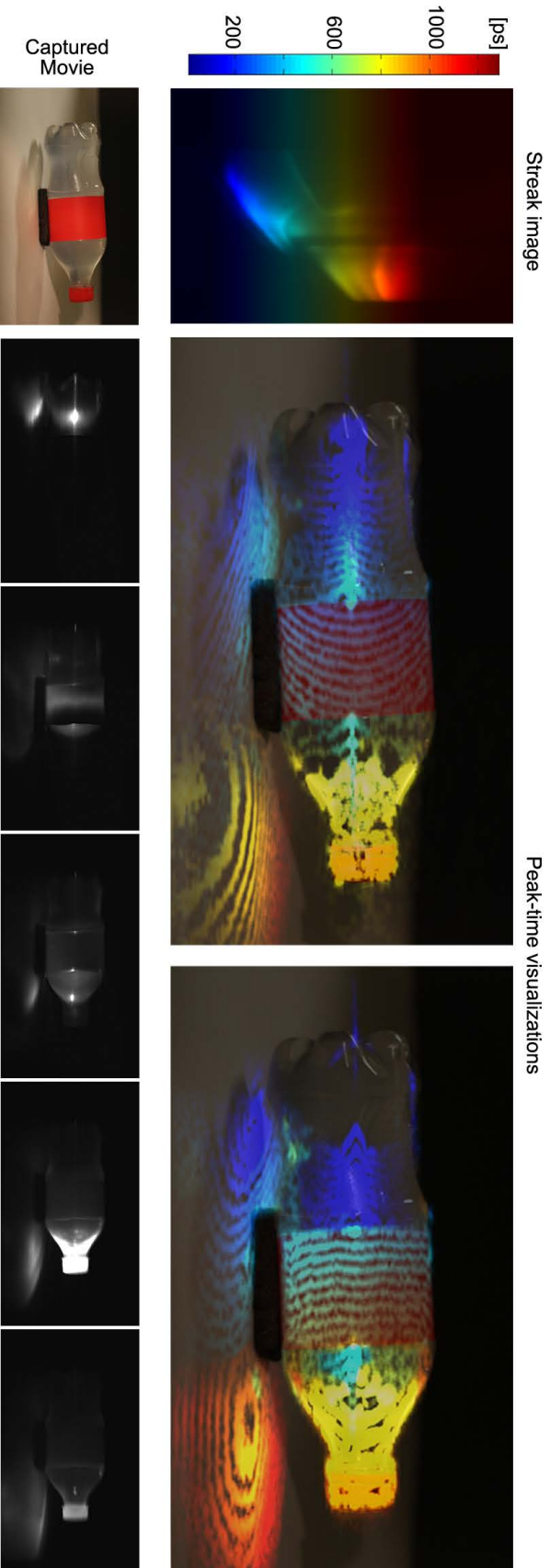
Figure 4.1: What does the world look like at the speed of light? Our new computational photography technique allows us to visualize light in ultra-slow motion, as it travels and interacts with objects in table-top scenes. We capture photons with an effective temporal resolution of less than 2 picoseconds per frame. Top row, left: a false color, single *streak image* from our sensor. Middle: time lapse visualization of the *bottle* scene, as directly reconstructed from sensor data. Right: time-unwarped visualization, taking into account the fact that the speed of light can no longer be considered infinite (see the main text for details). Bottom row: original scene through which a laser pulse propagates, followed by different frames of the complete reconstructed video. For this and other results in the paper, we refer the reader to the video included in the project page[a].

one half trillion frames per second. This allows us to see, for instance, a light pulse scattering inside a plastic bottle, or image formation in a mirror, as a *function of time*.

CHALLENGES   Developing such time-resolved system is a challenging problem for several reasons that are under-appreciated in conventional methods: (a) brute-force time exposures under 2 ps yield an impractical signal-to-noise (SNR) ratio; (b) suitable cameras to record 2D image sequences at this time resolution do not exist due to sensor bandwidth limitations; (c) comprehensible visualization of the captured time-resolved data is non-trivial; and (d) direct measurements of events appear warped in space-time, because the finite speed of light implies that the recorded light propagation delay depends on camera position relative to the scene.

CONTRIBUTIONS   Our main contribution is in addressing these challenges and creating a first prototype as follows:

- We exploit the statistical similarity of periodic light transport events to record multiple, ultrashort exposure times of one-dimensional views (Section 4.3).

- We introduce a novel hardware implementation to sweep the exposures across a vertical field of view, to build 3D space-time data volumes (Section 4.4).

- We create techniques for comprehensible visualization, including movies showing the dynamics of real-world light transport phenomena (including reflections, scattering, diffuse inter-reflections, or beam diffraction) and the notion of *peak-time*, which partially overcomes the low-frequency appearance of integrated global light transport (Section 4.5).

- We introduce a *time-unwarping* technique to correct the distortions in captured time-resolved information due to the finite speed of light (Section 4.6).

LIMITATIONS   Although not conceptual, our setup has several practical limitations, primarily due to the limited SNR of scattered light. Since the hardware elements in our system were originally designed for different purposes, it is not optimized for efficiency and suffers from low optical throughput (e.g., the detector is optimized for 500 nm visible light, while the infrared laser wavelength we use is 795 nm), and from dynamic range limitations. This lengthens the total recording time to approximately one hour. Furthermore, the scanning mirror, rotating continuously, introduces some blurring in the data along the scanned (vertical) dimension. Future optimized systems can overcome these limitations.

4.2   RELATED WORK

ULTRAFAST DEVICES   The fastest 2D continuous, real-time monochromatic camera operates at hundreds of nanoseconds per frame [53] (about $6 \cdot 10^6$ frames per second), with a spatial resolution of 200×200 pixels, less than one third of what we achieve. Avalanche photodetector (APD) arrays can reach temporal resolutions of several tens of picoseconds if they are used in a photon starved regime where only a single photon hits a detector within a time window of tens of nanoseconds [19]. Repetitive illumination techniques

used in incoherent LiDAR [214, 48] use cameras with typical exposure times on the order of hundreds of picoseconds [15, 21], two orders of magnitude slower than our system. Liquid nonlinear shutters actuated with powerful laser pulses have been used to capture single analog frames imaging light pulses at picosecond time resolution [32]. Other sensors that use a coherent phase relation between the illumination and the detected light, such as optical coherence tomography (OCT) [89], coherent LiDAR [250], light-in-flight holography [1], or white light interferometry [249], achieve femtosecond resolutions; however, they require light to maintain coherence (i.e., wave interference effects) during light transport, and are therefore unsuitable for indirect illumination, in which diffuse reflections remove coherence from the light. Simple streak sensors capture incoherent light at picosecond to nanosecond speeds, but are limited to a line or low resolution (20 × 20) square field of view [17, 93, 200, 48, 122, 181]. They have also been used as line scanning devices for image transmission through highly scattering turbid media, by recording the ballistic photons, which travel a straight path through the scatterer and thus arrive first on the sensor [76]. The principles that we develop in this paper for the purpose of transient imaging were first demonstrated by Velten et al. [225]. Recently, photonic mixer devices, along with nonlinear optimization, have also been used in this context [78].

Our system can record and reconstruct space-time world information of incoherent light propagation in free-space, table-top scenes, at a resolution of up to 672 × 1000 pixels and under 2 picoseconds per frame. The varied range and complexity of the scenes we capture allow us to visualize the *dynamics* of global illumination effects, such as scattering, specular reflections, interreflections, subsurface scattering, caustics, and diffraction.

TIME-RESOLVED IMAGING   Recent advances in time-resolved imaging have been exploited to recover geometry and motion around corners [186, 120, 223, 222, 59, 171] and albedo of from single view point [162]. But, none of them explored the idea of capturing videos of light in motion in direct view and have some fundamental limitations (such as capturing only third-bounce light) that make them unsuitable for the present purpose. Wu et al. [245] separate direct and global illumination components from time-resolved data captured with the system we describe in this paper, by analyzing the time profile of each pixel. In a recent publication [247], the authors present an analysis on transient light transport in frequency space, and show how it can be applied to bare-sensor imaging.

4.3   CAPTURING SPACE-TIME PLANES

We capture time scales orders of magnitude faster than the exposure times of conventional cameras, in which photons reaching the sensor at different times are integrated into a single value, making it impossible to observe ultrafast optical phenomena. The system described in this paper has an effective exposure time down to 1.85 ps; since light travels at 0.3 mm/ps, light travels approximately 0.5 mm between frames in our reconstructed movies.

SYSTEM: An ultrafast setup must overcome several difficulties in order to accurately measure a high-resolution (both in space and time) image. First, for an unamplified laser pulse, a single exposure time of less than 2 ps would not collect enough light, so the SNR would be unworkably low. As an example, for a table-top scene illuminated by a 100 W bulb, only about 1
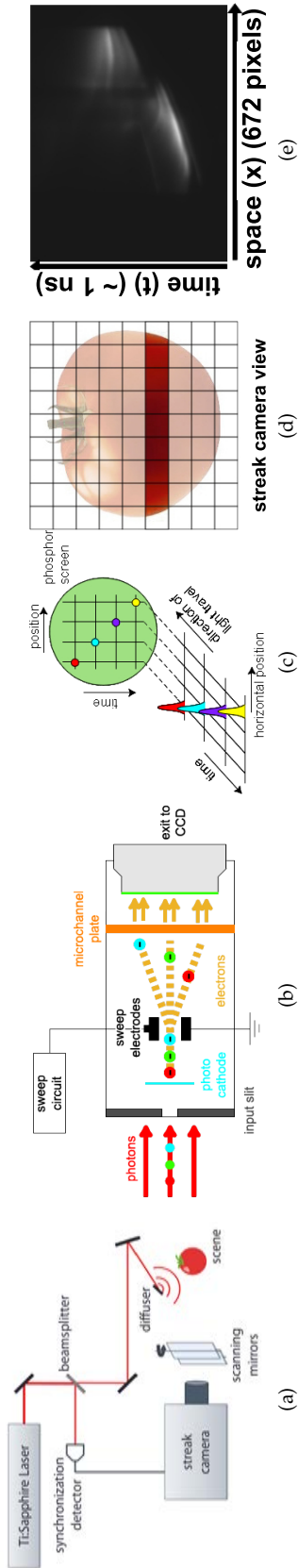
Figure 4.2: Our setup for capturing a single 1D space-time photo. (a) A laser beam strikes a diffuser, which converts the beam into a spherical energy front that illuminates the scene; a beamsplitter and a synchronization detector enable synchronization between the laser and the streak sensor. (b) After interacting with the scene, photons enter a horizontal slit in the camera and strike a photocathode, which generates electrons. These are deflected at different angles as they pass through a microchannel plate, by means of rapidly changing the voltage between the electrodes. (c) The CCD records the horizontal position of each pulse and maps its arrival time to the vertical axis, depending on how much the electrons have been deflected. (d) We focus the streak sensor on a single narrow scan line of the scene. (e) Sample image taken by the streak sensor. The horizontal axis (672 pixels) records the photons' spatial locations in the acquired scanline, while the vertical axis (1 nanosecond window in our implementation) codes their arrival time. Rotating the adjustable mirrors shown in (a) allows for scanning of the scene in the y-axis and generation of ultrafast 2D movies such as the one visualized in Figure 4.1. (Figures (a)-(d), credit: [47])
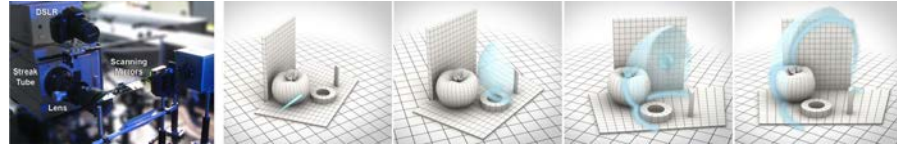
Figure 4.3: Left: Photograph of our ultrafast imaging system setup. The DSLR camera takes a conventional photo for comparison. Right: Time sequence illustrating the arrival of the pulse striking a diffuser, its transformation
into a spherical energy front, and its propagation through the scene. The
corresponding captured scene is shown in Figure 4.10 (top row).

photon on average would reach the sensor during a 2 ps open-shutter period.
Second, because of the time scales involved, synchronization of the sensor
and the illumination must be executed within picosecond precision. Third,
standalone streak sensors sacrifice the vertical spatial dimension in order
to code the time dimension, thus producing x-t images. As a consequence,
their field of view is reduced to a single horizontal line of view of the scene.

We solve these problems with our ultrafast imaging system, outlined in
Figure 4.2. (A photograph of the actual setup is shown in Figure 4.3 (left)).
The light source is a femtosecond (fs) Kerr lens mode-locked Ti:Sapphire
laser, which emits 50-fs with a center wavelength of 795 nm, at a repetition
rate of 75 MHz and average power of 500 mW. In order to see ultrafast
events in a scene with macro-scaled objects, we focus the light with a lens
onto a Lambertian diffuser, which then acts as a point light source and
illuminates the entire scene with a spherically-shaped pulse (see Figure 4.3
(right)). Alternatively, if we want to observe pulse propagation itself, rather
than the interactions with large objects, we direct the laser beam across the
field of view of the camera through a scattering medium (see the *bottle* scene
in Figure 4.1).

Because all the pulses are statistically identical, we can record the scattered light from many of them and integrate the measurements to average
out any noise. The result is a signal with a high SNR. To synchronize this
illumination with the streak sensor (Hamamatsu C5680 [71]), we split off a
portion of the beam with a glass slide and direct it onto a fast photodetector
connected to the sensor, so that, now, both detector and illumination operate
synchronously (see Figure 4.2 (a)).

CAPTURING SPACE-TIME PLANES: The streak sensor then captures an
x-t image of a certain scanline (i.e. a line of pixels in the horizontal dimension) of the scene with a space-time resolution of 672×512. The exact time
resolution depends on the amplification of an internal sweep voltage signal
applied to the streak sensor. With our hardware, it can be adjusted from 0.30
ps to 5.07 ps. Practically, we choose the fastest resolution that still allows for
capture of the entire duration of the event. In the streak sensor, a photocathode converts incoming photons, arriving from each spatial location in
the scanline, into electrons. The streak sensor generates the x-t image by
deflecting these electrons, according to the time of their arrival, to different
positions along the t-dimension of the sensor (see Figure 4.2(b) and 4.2(c)).
This is achieved by means of rapidly changing the sweep voltage between
the electrodes in the sensor. For each horizontal scanline, the camera records
a scene illuminated by the pulse and averages the light scattered by 4.5×10^8
pulses (see Figure 4.2(d) and 4.2(e)).
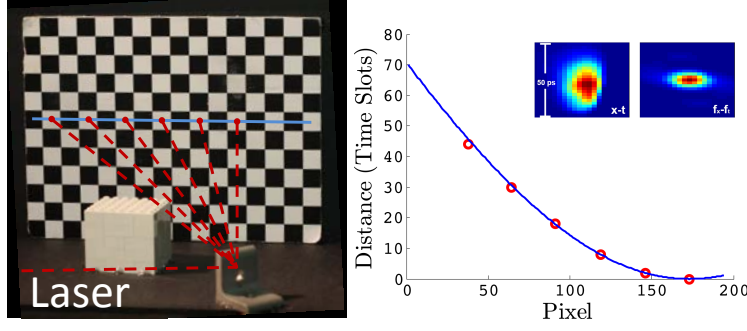
Figure 4.4: Performance validation of our system. Left: Measurement setup used to validate the data. We use a single streak image representing a line of the scene and consider the centers of the white patches because they are easily identified in the data. Right: Graph showing pixel position vs. total path travel time captured by the streak sensor (red) and calculated from measurements of the checkerboard plane position with a Faro digitizer arm (blue). Inset: PSF, and its Fourier transform, of our system.

PERFORMANCE VALIDATION   To characterize the streak sensor, we compare sensor measurements with known geometry and verify the linearity, reproducibility, and calibration of the time measurements. To do this, we first capture a streak image of a scanline of a simple scene: a plane being illuminated by the laser after hitting the diffuser (see Figure 4.4 (left)). Then, by using a Faro digitizer arm [37], we obtain the ground truth geometry of the points along that plane and of the point of the diffuser hit by the laser; this allows us to compute the total travel time per path (diffuser-plane-streak sensor) for each pixel in the scanline. We then compare the travel time captured by our streak sensor with the real travel time computed from the known geometry. The graph in Figure 4.4 (right) shows agreement between the measurement and calculation.

4.4   CAPTURING SPACE-TIME VOLUMES

Although the synchronized, pulsed measurements overcome SNR issues, the streak sensor still provides only a one-dimensional movie. Extension to two dimensions requires unfeasible bandwidths: a typical dimension is roughly 10^3 pixels, so a three-dimensional data cube has 10^9 elements. Recording such a large quantity in a 10^{-9} second (1 ns) time widow requires a bandwidth of 10^{18} byte/s, far beyond typical available bandwidths.

We solve this acquisition problem by again utilizing the synchronized repeatability of the hardware: A mirror-scanning system (two 9 cm × 13 cm mirrors, see Figure 4.3 (left)) rotates the camera's center of projection, so that it records horizontal slices of a scene sequentially. We use a computer-controlled, one-rpm servo motor to rotate one of the mirrors and consequently scan the field of view vertically. The scenes are about 25 cm wide and placed about 1 meter from the camera. With high gear ratios (up to 1:1000), the continuous rotation of the mirror is slow enough to allow the camera to record each line for about six seconds, requiring about one hour for 600 lines (our video resolution). We generally capture extra lines, above and below the scene (up to 1000 lines), and then crop them to match the aspect ratio of the physical scenes before the movie was reconstructed.

These resulting images are combined into one matrix, M_{ijk}, where $i = 1...672$ and $k = 1...512$ are the dimensions of the individual x-t streak im-
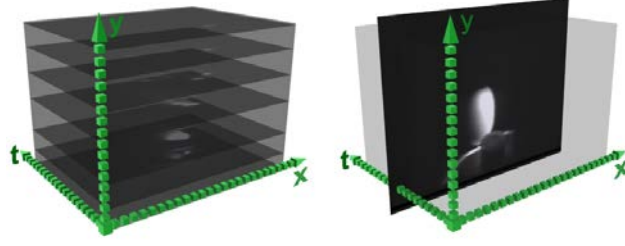
Figure 4.5: Left: Reconstructed x-y-t data volume by stacking individual x-t images (captured with the scanning mirrors). Right: An x-y slice of the data cube represents one frame of the final movie.

ages, and $j = 1...1000$ addresses the second spatial dimension y. For a given time instant k, the submatrix N_{ij} contains a two-dimensional image of the scene with a resolution of 672×1000 pixels, exposed for as short to 1.85 ps. Combining the x-t slices of the scene for each scanline yields a 3D x-y-t data volume, as shown in Figure 4.5 (left). An x-y slice represents one frame of the final movie, as shown in Figure 4.5 (right).

4.5 DEPICTING ULTRAFAST VIDEOS IN 2D

We have explored several ways to visualize the information contained in the captured x-y-t data cube in an intuitive way. First, contiguous N_{ij} slices can be played as the frames of a movie. Figure 4.1 (bottom row) shows a captured scene (*bottle*) along with several representative N_{ij} frames. (Effects are described for various scenes in Section 7.) However, understanding all the phenomena shown in a video is not a trivial task, and movies composed of x-y frames such as the ones shown in Figure 4.10 may be hard to interpret. Merging a static photograph of the scene from approximately the same point of view with the N_{ij} slices aids in the understanding of light transport in the scenes (see movies within the supplementary video). Although straightforward to implement, the high dynamic range of the streak data requires a nonlinear intensity transformation to extract subtle optical effects in the presence of high intensity reflections. We employ a logarithmic transformation to this end.

We have also explored single-image methods for intuitive visualization of full space-time propagation, such as the color-coding in Figure 4.1 (right), which we describe in the following paragraphs.

INTEGRAL PHOTO FUSION   By integrating all the frames in novel ways, we can visualize and highlight different aspects of the light flow in one photo. Our photo fusion results are calculated as $N_{ij} = \sum w_k M_{ijk}, \{k = 1..512\}$, where w_k is a weighting factor determined by the particular fusion method. We have tested several different methods, of which two were found to yield the most intuitive results: the first one is *full fusion*, where $w_k = 1$ for all k. Summing all frames of the movie provides something resembling a black and white photograph of the scene illuminated by the laser, while showing time-resolved light transport effects. An example is shown in Figure 4.6 (left) for the *alien* scene. (More information about the scene is given in Section 4.7.) A second technique, *rainbow fusion*, takes the fusion result and assigns a different RGB color to each frame, effectively color-coding the temporal dimension. An example is shown in Figure 4.6 (middle).

| Full Fusion | Rainbow Fusion | Peak Time |
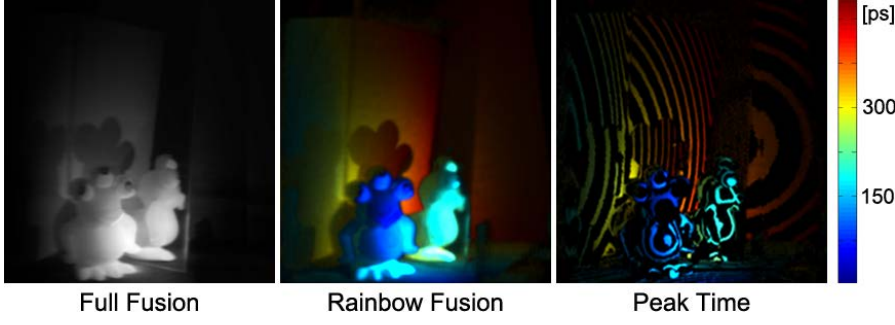
Figure 4.6: Three visualization methods for the *alien* scene. From left to right, more sophisticated methods provide more information and an easier interpretation of light transport in the scene.

PEAK TIME IMAGES   The inherent integration in fusion methods, though often useful, can fail to reveal the most complex or subtle behavior of light. As an alternative, we propose peak time images, which illustrate the time evolution of the *maximum* intensity in each frame. For each spatial position (i, j) in the x-y-t volume, we find the peak intensity along the time dimension, and keep information within two time units to each side of the peak. All other values in the streak image are set to zero, yielding a more sparse space-time volume. We then color-code time and sum up the x-y frames in this new sparse volume, in the same manner as in the *rainbow fusion* case but use only every 20th frame in the sum to create black lines between the equi-time paths, or isochrones. This results in a map of the propagation of maximum intensity contours, which we term *peak time image*. These color-coded isochronous lines can be thought of intuitively as propagating energy fronts. Figure 4.6 (right) shows the peak time image for the *alien* scene, and Figure 4.1 (top, middle) shows the captured data for the *bottle* scene depicted using this visualization method. As explained in the next section, this visualization of the bottle scene reveals significant light transport phenomena that could not be seen with the rainbow fusion visualization.

4.6 TIME UNWARPING

Visualization of the captured movies (Sections 4.5 and 4.7) reveals results that are counter-intuitive to theoretical and established knowledge of light transport. Figure 4.1 (top, middle) shows a peak time visualization of the *bottle* scene, where several abnormal light transport effects can be observed: (1) the caustics on the floor, which propagate towards the bottle, instead of away from it; (2) the curved spherical energy fronts in the label area, which should be rectilinear as seen from the camera; and (3) the pulse itself being located behind these energy fronts, when it would need to precede them. These are due to the fact that usually light propagation is assumed to be infinitely fast, so that events in world space are assumed to be detected simultaneously in camera space. In our ultrafast photography setup, however, this assumption no longer holds, and the finite speed of light becomes a factor: we must now take into account the time delay between the occurrence of an event and its detection by the camera sensor.

We therefore need to consider two different time frames, namely *world time* (when events happen) and *camera time* (when events are detected). This duality of time frames is explained in Figure 4.7: light from a source hits a surface first at point $P_1 = (i_1, j_1)$ (with (i, j) being the x-y pixel coordinates
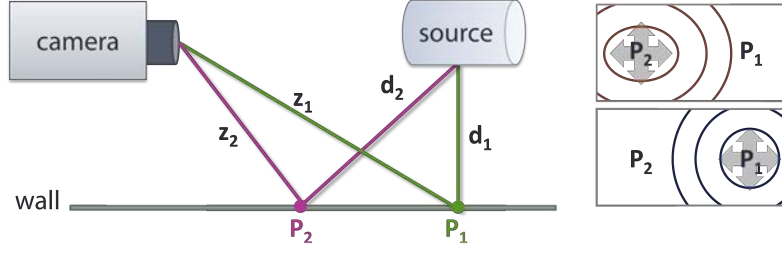
Figure 4.7: Understanding reversal of events in captured videos. *Left:* Pulsed light scatters from a source, strikes a surface (e.g., at P_1 and P_2), and is then recorded by a sensor. Time taken by light to travel distances $z_1 + d_1$ and $z_2 + d_2$ is responsible for the existence of two different time frames and the need of computational correction to visualize the captured data in the world time frame. *Right:* Light appears to be propagating from P_2 to P_1 in camera time (before unwarping), and from P_1 to P_2 in world time, once time-unwarped. Extended, planar surfaces will intersect constant-time paths to produce either elliptical or circular fronts.
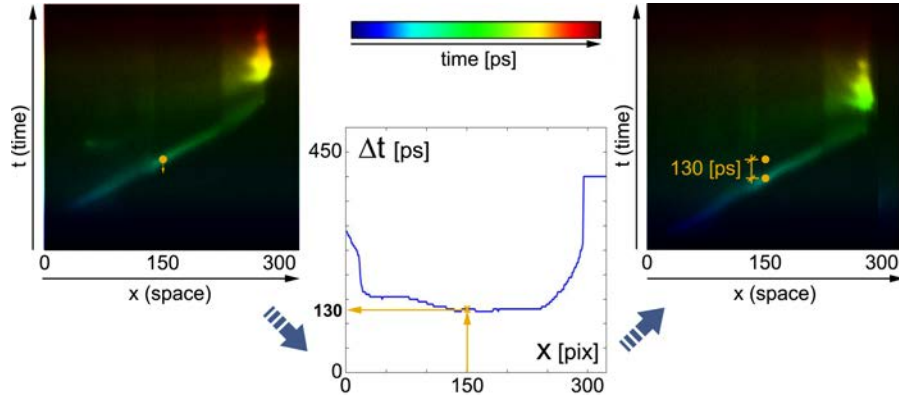
Figure 4.8: Time unwarping in 1D for a streak image (x-t slice). Left: captured streak image; shifting the time profile down in the temporal dimension by Δt allows for the correction of path length delay to transform between time frames. Center: the graph shows, for each spatial location x_i of the streak image, the amount Δt_i that point has to be shifted in the time dimension of the streak image. Right: resulting time-unwarped streak image.

of a scene point in the x-y-t data cube), then at the farther point $P_2 = (i_2, j_2)$, but the reflected light is captured in the reverse order by the sensor, due to different total path lengths ($z_1 + d_1 > z_2 + d_2$). Generally, this is due to the fact that, for light to arrive at a given time instant t_0, all the rays from the source, to the wall, to the camera, must satisfy $z_i + d_i = ct_0$, so that isochrones are elliptical. Therefore, although objects closer to the source receive light earlier, they can still lie on a higher-valued (later-time) isochrone than farther ones.

In order to visualize all light transport events as they have occurred (not as the camera captured them), we transform the captured data from camera time to world time, a transformation which we term *time unwarping*. Mathematically, for a scene point $P = (i, j)$, we apply the following transformation:

$$t'_{ij} = t_{ij} + \frac{z_{ij}}{c/\eta} \tag{4.1}$$

Figure 4.9: Time unwarping for the *bottle* scene, containing a scattering medium. From left to right: a frame of the video without correction, where the energy front appears curved; the same frame after time-unwarping with respect to distance to the camera z_{ij}; the shape of the energy front is now correct, but it still appears before the pulse; the same frame, time-unwarped taking also scattering into account.

where t'_{ij} and t_{ij} represent camera and world times respectively, c is the speed of light in vacuum, η the index of refraction of the medium, and z_{ij} is the distance from point P to the camera. For our table-top scenes, we measure this distance with a Faro digitizer arm, although it could be obtained from the data and the known position of the diffuser, as the problem is analogous to that of bi-static LiDAR. We can thus define light travel time from each point (i, j) in the scene to the camera as $\Delta t_{ij} = t'_{ij} - t_{ij} = z_{ij}/(c/\eta)$. Then, time unwarping effectively corresponds to offsetting data in the x-y-t volume along the time dimension, according to the value of Δt_{ij} for each of the (i, j) points, as shown in Figure 4.8.

In most of the scenes, we only have propagation of light through air, for which we take $\eta \approx 1$. For the *bottle* scene, we assume that the laser pulse travels along its longitudinal axis at the speed of light, and that only a single scattering event occurs in the liquid inside. We take $\eta = 1.33$ as the index of refraction of the liquid and ignore refraction at the bottle's surface. A step-by-step unwarping process is shown in Figure 4.9 for a frame (i.e. x-y image) of the *bottle* scene. Our unoptimized Matlab code runs at about 0.1 seconds per frame. A time-unwarped peak-time visualization of the whole of this scene is shown in Figure 4.1 (right). Notice how now the caustics originate from the bottle and propagate outward, energy fronts along the label are correctly depicted as straight lines, and the pulse precedes related phenomena, as expected.

4.7 CAPTURED SCENES

We have used our ultrafast photography setup to capture interesting light transport effects in different scenes. Figure 4.10 summarizes them, showing representative frames and peak time visualizations. The exposure time for our scenes is between 1.85 ps for the *crystal* scene, and 5.07 ps for the *bottle* and *tank* scenes, which required imaging a longer time span for better visualization. Please refer to the supplementary video[1] to watch the reconstructed movies. Overall, observing light in such slow motion reveals both subtle and key aspects of light transport. We provide here brief descriptions of the light transport effects captured in the different scenes.

bottle This scene is shown in Figure 1 (bottom row), and has been used to introduce time-unwarping. A plastic bottle, filled with water diluted with milk, is directly illuminated by the laser pulse, entering through the bottom of the bottle along its longitudinal axis. The pulse scatters inside the

---

1 http://giga.cps.unizar.es/~ajarabo/pubs/femtoSIG2013/downloads/velten_sig13_movie.mov
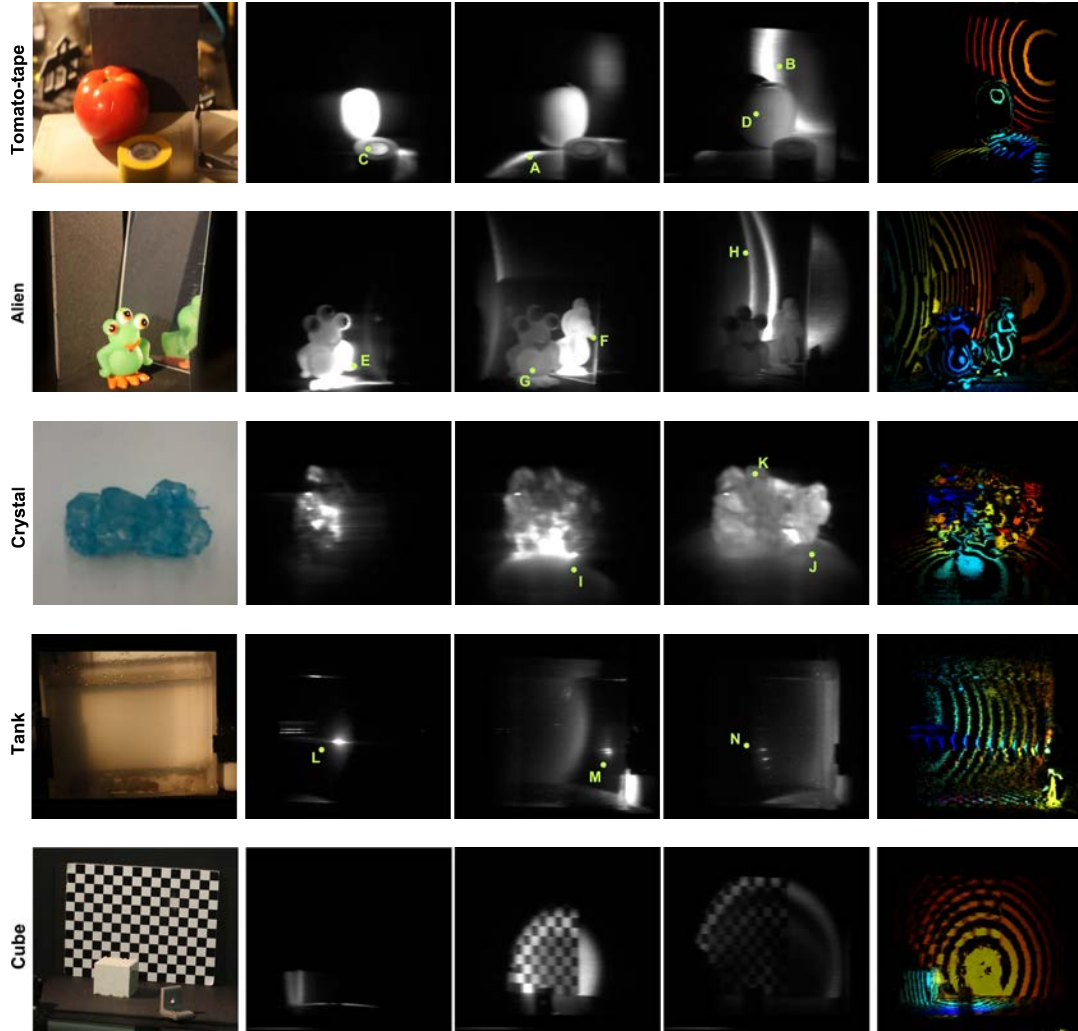
Figure 4.10: More scenes captured with our setup (refer to Figure 1 for the *bottle* scene). For each scene, from left to right: photograph of the scene (taken with a DSLR camera), a series of representative frames of the reconstructed movie, and peak time visualization of the data. Please refer to the videos in the project page for the full movies. Note that the viewpoint varies slightly between the DSLR and the streak sensor.

liquid; we can see the propagation of the wavefronts. The geometry of the bottle neck creates some interesting lens effects, making light look almost like a fluid. Most of the light is reflected back from the cap, while some is transmitted or trapped in subsurface scattering phenomena. Caustics are generated on the table.

tomato-tape   This scene shows a tomato and a tape roll, with a wall behind them. The propagation of the spherical wavefront, after the laser pulse hits the diffuser, can be seen clearly as it intersects the floor and the back wall (A, B). The inside of the tape roll is out of the line of sight of the light source and is not directly illuminated. It is illuminated later, as indirect light scattered from the first wave reaches it (C). Shadows become visible only after the object has been illuminated. The more opaque tape darkens quickly after the light front has passed, while the tomato continues glowing for a longer time, indicative of stronger subsurface scattering (D).

alien   A toy alien is positioned in front of a mirror and wall. Light interactions in this scene are extremely rich, due to the mirror, the multiple interreflections, and the subsurface scattering in the toy. The video shows how the reflection in the mirror is actually formed: direct light first reaches the toy, but the mirror is still completely dark (E); eventually light leaving the toy reaches the mirror, and the reflection is dynamically formed (F). Subsurface scattering is clearly present in the toy (G), while multiple direct and indirect interactions between the wall and the mirror can also be seen (H).

crystal   A group of sugar crystals is directly illuminated by the laser from the left, acting as multiple lenses and creating caustics on the table (I). Part of the light refracted on the table is reflected back to the candy, creating secondary caustics on the table (J). Additionally, scattering events are visible within the crystals (K).

tank   A reflective grating is placed at the right side of a tank filled with milk diluted in water. The grating is taken from a commercial spectrometer, and consists of an array of small, equally spaced rectangular mirrors. The grating is blazed: mirrors are tilted to concentrate maximum optical power in the first order diffraction for one wavelength. The pulse enters the scene from the left, travels through the tank (L), and strikes the grating. The grating reflects and diffracts the beam pulse (M). The different orders of the diffraction are visible traveling back through the tank (N). As the figure (and the supplementary movie) shows, most of the light reflected from the grating propagates at the blaze angle.

cube   A very simple scene composed of a cube in front of a wall with a checkerboard pattern. The simple geometry allows for a clear visualization and understanding of the propagation of wavefronts.

4.8   CONCLUSIONS AND FUTURE WORK

Our research fosters new computational imaging and image processing opportunities by providing incoherent time-resolved information at ultrafast temporal resolutions. We hope our work will inspire new research in computer graphics and computational photography, by enabling forward and inverse analysis of light transport, allowing for full scene capture of hidden

geometry and materials, or for relighting photographs. To this end, captured movies and data of the scenes shown in this chapter are available at the project page[2]. This exploitation, in turn, may influence the rapidly emerging field of ultrafast imaging hardware.

The system could be extended to image in color by adding additional pulsed laser sources at different colors or by using one continuously tunable optical parametric oscillator (OPO). A second color of about 400 nm could easily be added to the existing system by doubling the laser frequency with a nonlinear crystal (about $1000). The streak tube is sensitive across the entire visible spectrum, with a peak sensitivity at about 450 nm (about five times the sensitivity at 800 nm). Scaling to bigger scenes would require less time resolution and could therefore simplify the imaging setup. Scaling should be possible without signal degradation, as long as the camera aperture and lens are scaled with the rest of the setup. If the aperture stays the same, the light intensity needs to be increased quadratically to obtain similar results.

Beyond the ability of the commercially available streak sensor, advances in optics, material science, and compressive sensing may bring further optimization of the system, which could yield increased resolution of the captured x-t streak images. Nonlinear shutters may provide an alternate path to femto-photography capture systems. However, nonlinear optical methods require exotic materials and strong light intensities that can damage the objects of interest (and must be provided by laser light). Further, they often suffer from physical instabilities.

We believe that mass production of streak sensors can lead to affordable systems. Also, future designs may overcome the current limitations of our prototype regarding optical efficiency. Future research can investigate other ultrafast phenomena such as propagation of light in anisotropic media and photonic crystals, or may be used in applications such as scientific visualization (to understand ultra-fast processes), medicine (to reconstruct subsurface elements), material engineering (to analyze material properties), or quality control (to detect faults in structures). This could provide radically new challenges in the realm of computer graphics. Graphics research can enable new insights via comprehensible simulations and new data structures to render light in motion. For instance, in Chapter 6 we show relativistic rendering techniques developed using this data, where the common assumption of constant irradiance over the surfaces does no longer hold [102, 103]. It may also allow a better understanding of scattering, and may lead to new physically valid models, as well as spawn new art forms.

---

2 http://giga.cps.unizar.es/~ajarabo/pubs/femtoSIG2013/

A FRAMEWORK FOR TRANSIENT RENDERING

In this chapter we present a framework for rendering time-resolved light transport. It originated to address the need of simulation tools helping on the development of the incipient field of *transient imaging*. We first introduce the *transient path integral* framework, formally describing light transport in transient state, and then analyze the difficulties arising when considering the light's time-of-flight in the simulation (rendering) of images and videos. We propose a novel density estimation technique that allows reusing sampled paths to reconstruct time-resolved radiance, and devise new sampling strategies that take into account the distribution of radiance along time in participating media. This work was published in ACM Transactions on Graphics and presented at SIGGRAPH Asia 2014.

5.1 INTRODUCTION

One of the most general assumptions in computer graphics is to consider the speed of light to be infinite, leading to the simulation of light transport in steady state. This is a reasonable assumption, since most of the existing imaging hardware is very slow compared to the speed of light. Light transport in steady state has been extensively investigated in computer graphics (e.g. Dutré et al. [34], Gutierrez et al. [63], Křivánek et al. [132]), including for instance the gradient [182, 108] or frequency [33] domains. In contrast, work in the temporal domain has been mainly limited to simulating motion blur [163] or time-of-flight imaging [123].

We introduce in this work a formal framework for *transient rendering*, where we lift the assumption of an infinite speed of light. While different works have looked into transient rendering [205, 96, 3], they have approached the problem by proposing straight forward extensions of traditional steady-state algorithms, which are not adequate for *efficient* transient rendering for a variety of reasons. Firstly, the addition of the extra sampling domain given by the temporal dimension dramatically increases the convergence time of steady state rendering algorithms. Moreover, by extending the well-accepted path integral formulation [218], we observe that paths contributing to each frame form a near-delta manifold *in time*, which makes sampling almost impossible. We solve this issue by devising new sampling strategies that improve the distribution of samples *along the temporal domain*, and a new density estimation technique that allows reconstructing the signal along time from such samples.

Our paper presents valuable insight apart from rendering applications. Recent advances in time-resolved imaging are starting to provide novel solutions to open problems, such as reconstructing hidden geometry [223] or BRDFs [162], recovering depth of transparent objects [111], or even visualizing the propagation of light [224]. Despite these breakthroughs in technology, there is currently a lack of tools to efficiently simulate and analyze
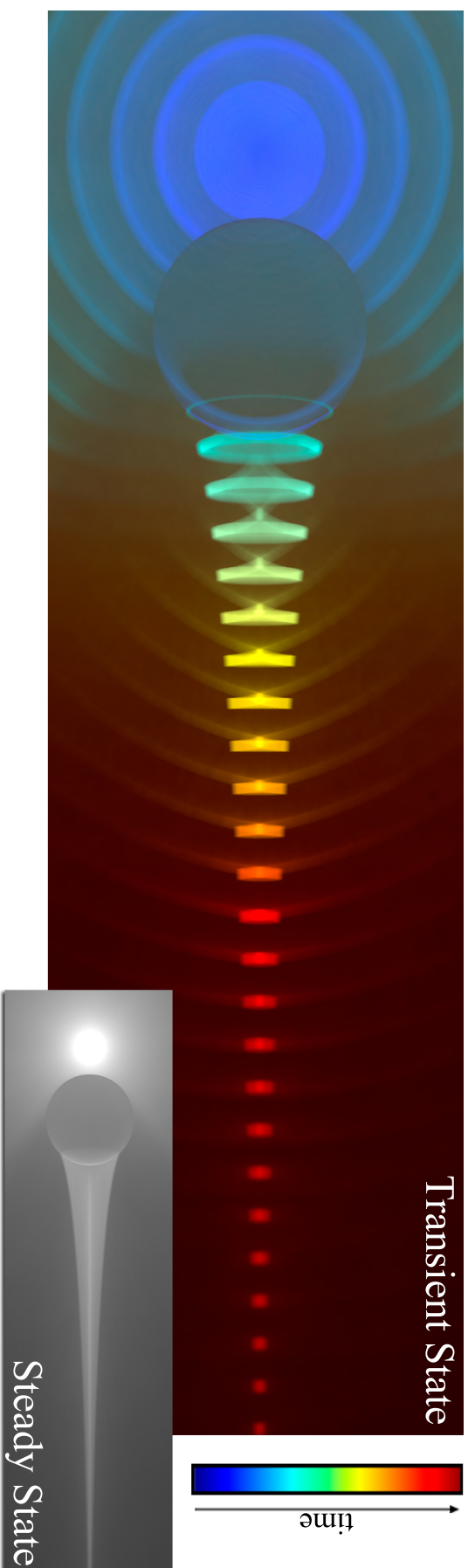
Figure 5.1: Our transient rendering framework allows time-resolved visualizations of light propagation. The caustic wavefront produced by the spherical lens above is distorted by the lens and also delayed due to the longer optical path traversed within the sphere. Note that in this scene we omit the propagation time of the last segment (from the scene to the camera).

transient light transport. This would not only be beneficial for the graphics and vision communities, but it could open up a novel analysis-by-synthesis approach for applications in fields like optical imaging, material engineering or biomedicine as well. In addition, our framework can become instrumental in teaching the complexities of light transport [108], as well as visualizing in detail some of its most cumbersome aspects, such as the formation of caustics, birefringence, or the temporal evolution of chromatic dispersion.

In particular, in this work we make the following contributions:

- Establishing a theoretical framework for rendering in transient state, based on the path integral formulation and including propagation in free space as well as scattering on both surfaces and in media (Section 5.4). This allows us to analyze the main challenges in transient rendering (Section 5.4.1).
- Developing a progressive kernel-based density estimation technique for path reuse that significantly improves the reconstruction of time-resolved radiance (Section 5.5).
- Devising new sampling techniques for participating media to uniformly sample in the temporal domain, that complement traditional radiance-based sampling (Section 5.6).
- Providing time-resolved simulations of several light transport phenomena which are impossible to see in steady state (Section 5.7).

5.2 RELATED WORK

TRANSIENT RADIATIVE TRANSFER.   With advances in laser technology, capable of producing pulses of light in the order of a few femtoseconds, *transient* radiative transfer gained relevance in fields like optical imaging, material engineering or biomedicine. Many numerical strategies have been proposed, including Monte Carlo simulations, discrete ordinate methods, integral equation models or finite volume methods [155, 257, 259]. Often, these methods are applied on simplified scenarios with a particular application in mind, but a generalized framework has not yet been adopted.

ULTRA-FAST IMAGING.   Several recent advances in ultra-fast imaging have found direct applications in computer graphics and vision. Raskar and Davis [186] introduce the basic theoretical framework in light transport analysis that would later lead to a number of practical applications, such as reconstruction of hidden geometry [120, 223] or reflectance acquisition [162]. Velten et al. [225, 224] have recently presented *femto-photography*, a technique that allows capturing time-resolved videos with an effective exposure time of one picosecond per frame, using a streak camera. Heide et al. [78] later propose a cheaper setup using Photonic Mixing Devices (PMDs), while sacrificing temporal and spatial resolution. Kadambi and colleagues [111] address multi path interference in time-of-flight sensors by recovering time profiles as a sequence of impulses, allowing them to recover depth from transparent objects.

ANALYSIS OF TIME-RESOLVED LIGHT TRANSPORT.   Wu et al. [247] analyze the propagation of light in the frequency domain, and show how the cross-dimensional transfer of information between the temporal and frequency domains can be applied to bare-sensor imaging. Later, Wu et al. [245] used time-of-flight imaging to approximately decompose light transport into its different components of direct, indirect and subsurface illumination,

by observing the temporal profiles at each pixel. Lin and colleagues [137] perform a frequency-domain analysis of multifrequency time-of-flight cameras. Recently, O'Toole and colleagues [168] derived transient light transport as a linear operator, as opposed to our formulation in ray space, and showed how to combine the generation and acquisition of transient light transport for scene analysis. In this regard, our work can be seen as complementary: we provide a simulation (rendering) framework, suitable for an analysis-by-synthesis approach to exploring novel ideas and applications, and to help better understand the mechanisms of light transport.

TRANSIENT RENDERING.   The term *transient rendering* was first coined by Smith et al. [205]. In their work, the authors generalize the rendering equation as a recursive operator including propagation of light at finite speed. The model provides a solid theoretical background for time-of-flight, computer vision applications, but does not provide a practical framework for transient rendering of global illumination. Keller et al. [114] develop a time-of-flight sensor simulation, modeling the behavior of PMDs. These works are again geared towards time-of-flight applications; moreover, they are limited to surfaces, not taking into account the presence of participating media. Simulation of relativistic effects [238, 102, 103] could also potentially benefit from our transient rendering framework, as shown later on Chapter 6.

Some recent works in computer graphics make use of transient state information: d'Eon and Irving [29] quantize light propagation into a set of states, and model the transient state at each instant using Gaussians with variance proportional to time. These Gaussians are then integrated into the final image. The wave-based approach by Musbach et al. [158] uses the Finite Difference Time Domain (FDTD) method to obtain a solution for Maxwell's equations, rendering complex effects like diffraction. In all these cases, however, the main goal is to render steady state images, not to analyze the propagation of light itself. Jarabo [96] showed transient rendering results based on photon mapping and time-dependent density estimation, but limited to surfaces in the absence of participating media. Last, Ament et al. [3] include time into the Radiative Transfer Equation in order to account for a continuously-varying indices of refraction in participating media, though they do not introduce efficient techniques for transient rendering.

ACOUSTIC RENDERING.   Our work is somewhat related to the field of acoustic rendering [45]. Traditional light rendering techniques have been adapted to sound rendering, such as photon (*phonon*) mapping [8] or precomputed *acoustic* radiance transfer [5]. Closest to our approach, the work by Siltanen et al. [202] extends the radiosity method to include propagation delays due to the finite, though much slower, speed of sound. As opposed to us, they use finite elements methods to compute sound transport, do not handle participating media and do not propose sampling techniques for uniform temporal sample distribution.

5.3 BACKGROUND

Here we introduce our notation and review the classic (steady state) path integral formulation, as well as the photon mapping algorithms and its progressive variant. The former will serve as the basis for the definition of *transient path integral* (Section 5.4), while the second closely relates with the
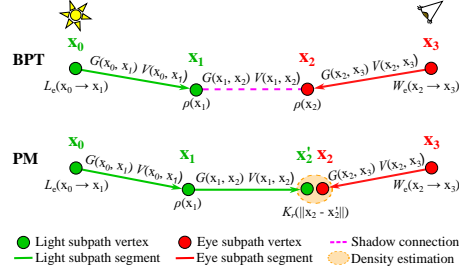
Figure 5.2: Schematic description of bidirectional path tracing (BPT, top) and photon mapping (PM, bottom). In both algorithms, an eye and a light subpath are traced from the eye and the light source respectively; these two subpaths are then connected to form a full path, via deterministic shadow connection in the case of bidirectional path tracing, and via an additional random segment and density estimation in photon mapping (Figure after Georgiev et al. [49]).

path reuse technique described in Section 5.5, which as we will see, its crucial for transient rendering.

5.3.1 *Path Integral*

In the path integral formulation [218, 172], the image pixel intensity I is computed as an integral over the space of light transport paths Ω:

$$I = \int_{\Omega} f(\bar{\mathbf{x}}) \, \mathrm{d}\mu(\bar{\mathbf{x}}), \tag{5.1}$$

where $\bar{\mathbf{x}} = \mathbf{x}_0 \ldots \mathbf{x}_k$ represents the spatial coordinates of the $k+1$ vertices of a length-k path with $k \geq 1$ segments. Vertex $\mathbf{x}_0$ lies on a light source, $\mathbf{x}_k$ lies on the camera sensor, and $\mathbf{x}_1 \ldots \mathbf{x}_{k-1}$ are intermediate scattering vertices. The differential measure $\mathrm{d}\mu(\bar{\mathbf{x}})$ denotes area integration for surfaces vertices and volume integration for media vertices. The path contribution function $f(\bar{\mathbf{x}})$ is the product of the emitted radiance L_e, path throughput $\mathfrak{T}$, and sensor importance W_e:

$$f(\bar{\mathbf{x}}) = L_\mathrm{e}(\mathbf{x}_0 \to \mathbf{x}_1) \, \mathfrak{T}(\bar{\mathbf{x}}) \, W_\mathrm{e}(\mathbf{x}_{k-1} \to \mathbf{x}_k). \tag{5.2}$$

The path throughput is itself the product of the scattering function ρ for the inner path vertices and the geometry G and visibility V terms for path segments:

$$\mathfrak{T}(\bar{\mathbf{x}}) = \left[\prod_{i=1}^{k-1} \rho(\mathbf{x}_i) \right] \left[\prod_{i=0}^{k-1} G(\mathbf{x}_i, \mathbf{x}_{i+1}) V(\mathbf{x}_i, \mathbf{x}_{i+1}) \right]. \tag{5.3}$$

For a path segment $\mathbf{xy}$, we have $G(\mathbf{x}, \mathbf{y}) = \frac{D(\mathbf{x} \to \mathbf{y}) D(\mathbf{y} \to \mathbf{x})}{\|\mathbf{x} - \mathbf{y}\|^2}$, where $D(\mathbf{x} \to \mathbf{y}) = |n_\mathbf{x} \cdot \omega_{\mathbf{xy}}|$ if $\mathbf{x}$ is on a surface and $D(\mathbf{x} \to \mathbf{y}) = 1$ if $\mathbf{x}$ is in a medium, and likewise for $D(\mathbf{y} \to \mathbf{x})$. Here $n_\mathbf{x}$ is the surface normal at $\mathbf{x}$ and $\omega_{\mathbf{xy}}$ is a unit-length vector from $\mathbf{x}$ to $\mathbf{y}$. We assume that V is a *fractional* visibility function accounting for transmittance within media in addition to binary visibility for opaque objects. For path segment $\mathbf{xy}$, it is given by $V(\mathbf{x}, \mathbf{y}) = \exp\left(-\int_0^{\|\mathbf{x} - \mathbf{y}\|} \sigma_t(\mathbf{x} + t\omega_{\mathbf{xy}}) \, \mathrm{d}t\right)$ where σ_t is the extinction coeffi-

cient of the medium or a delta function at the boundary of opaque objects. The scattering kernel at each vertex is defined as:

$$\rho(\mathbf{x}_i) = \begin{cases} \rho_s(\mathbf{x}_{i-1} \to \mathbf{x}_i \to \mathbf{x}_{i+1}) & \mathbf{x}_i \text{ on surface,} \\ \rho_p(\mathbf{x}_{i-1} \to \mathbf{x}_i \to \mathbf{x}_{i+1})\sigma_s(\mathbf{x}_i) & \mathbf{x}_i \text{ in medium,} \end{cases} \tag{5.4}$$

where σ_s is the scattering coefficient in the medium, and ρ_s and ρ_p are the surface BSDF and phase function respectively.

Monte Carlo solutions approximate the path integral as a Monte Carlo estimator:

$$\langle I \rangle = \frac{1}{n} \sum_{j=1}^{n} \frac{f(\overline{\mathbf{x}}_j)}{p(\overline{\mathbf{x}}_j)}, \tag{5.5}$$

that averages the contribution of n random paths $\overline{\mathbf{x}}_j$, sampled with a probability distribution function (pdf) $p(\overline{\mathbf{x}}_j)$, which is given by the combined probability density of each of the vertex locations $p(\overline{\mathbf{x}}) = p(\mathbf{x}_0...\mathbf{x}_k)$. The probability density of the path is determined by the sampling technique used to obtain the path: for example, *bidirectional path tracing* (BPT) [133, 219] independently generates a subpath $\overline{\mathbf{x}}_w$ from the eye with pdf $p(\overline{\mathbf{x}}_w)$ and a subpath $\overline{\mathbf{x}}_l$ from the light with pdf $p(\overline{\mathbf{x}}_l)$. These are then (optionally) connected using a shadow ray to build the full path $\overline{\mathbf{x}}$ with pdf $p(\overline{\mathbf{x}}) = p(\overline{\mathbf{x}}_l)p(\overline{\mathbf{x}}_w)$ (see Figure 5.2, top).

5.3.2 *Photon Mapping*

Photon mapping (PM) [109] is an efficient and robust two-pass global illumination algorithm. In the first pass (*photon tracing*), light subpaths are traced from the light sources, and their vertices hitting a diffuse surface are stored in a data structure called the photon map, which represents incoming flux.

In the second pass (*radiance estimation*, see Figure 5.2,bottom), PM estimates the reflected radiance at a point $\mathbf{x}$ in direction ω_o using density estimation from the M nearest photons as:

$$\widehat{L_o}(\mathbf{x}, \omega_o) = \frac{1}{M} \sum_{j=1}^{M} K_R(\|\mathbf{x} - \mathbf{x}_j\|)\gamma_j \tag{5.6}$$

where K_R is the spatial smoothing kernel with bandwidth R, and $\mathbf{x}_j$ and γ_j are the position and contribution of photon j. Note that γ_j is a function of the light subpath of the photon $\overline{\mathbf{x}}_{l,j}$, and is computed as: $\gamma_j = \rho(\omega_j \to \mathbf{x} \to \omega_o)L_e(\mathbf{x}_{0,j} \to \mathbf{x}_{1,j})\mathfrak{T}(\overline{\mathbf{x}}_{l,j})/p(\overline{\mathbf{x}}_{l,j})$, with ω_j the incoming direction of the photon j. PM successfully handles difficult light paths, at the price of introducing bias. This means that the estimated radiance in Equation (5.6) can be written as $\widehat{L_o} = L_o + \epsilon$, where L_o is the actual radiance and ϵ represents the error introduced by the density estimation. An important property of PM is that, although biased, it is consistent, meaning that the bias vanishes in the limit using an infinite number of photons M and a kernel K_R with differential bandwidth dR. This is obviously not a practical solution, so several improvements have been proposed over the last few years [65].

PROGRESSIVE PHOTON MAPPING   Progressive photon mapping (PPM) [68, 67, 121] is a multipass variant of photon mapping that allows handling difficult light paths without having to store an infinite number of photons in

the photon map to eliminate bias. The method progressively traces photons, estimates radiance and discards photons, updating the results at each step while ensuring convergence in the limit. Since a new photon map is computed at each iteration, this is equivalent to performing different independent samples on the radiance. The approximated pixel measure $\langle I_n \rangle$ in pass n is computed as:

$$\langle I_n \rangle = \frac{1}{n} \sum_{j=1}^{n} \Psi_j \widehat{L_o}(\mathbf{x}_j, \omega_{o,j}) \tag{5.7}$$

where $\Psi_j = \mathfrak{T}(\overline{\mathbf{x}}_{w,j}) / p(\overline{\mathbf{x}}_{w,j}) W_e(\mathbf{x}_{k,j} \to \mathbf{x}_{k-1,j})$ is the eye subpath contribution to $\mathbf{x}$.

Knaus and Zwicker [121] showed that this estimation is consistent if both the variance $\mathrm{Var}[\epsilon_n]$ and expected value $\mathrm{E}[\epsilon_n]$ of the error ϵ_n vanish as $n \to \infty$:

$$\mathrm{Var}[\epsilon_n] \to 0 \quad \Longrightarrow \quad \mathrm{Var}[\langle I_n \rangle] \to 0 \tag{5.8}$$

$$\mathrm{E}[\epsilon_n] \to 0 \quad \Longrightarrow \quad \mathrm{E}[\langle I_n \rangle] \to I. \tag{5.9}$$

To accomplish this, the bias $\mathrm{E}[\epsilon_j]$ is reduced in each iteration j by progressively reducing the bandwidth R of K_R, while allowing the $\mathrm{Var}[\epsilon_j]$ at each iteration to increase as:

$$\frac{\mathrm{Var}[\epsilon_{j+1}]}{\mathrm{Var}[\epsilon_j]} = \frac{R_j^2}{R_{j+1}^2} = \frac{j+1}{j+\alpha} \tag{5.10}$$

Note that this radius redution is valid for surfaces, for volumetric density estimation of photon points in media it is a function of R^3. The user parameter $\alpha \in (0,1)$ controls how much the variance is allowed to increase in each iteration. This value determines the trade-off between the reduction of bias and radiance [121], and its choice has a dramatic effect on the convergence rate of the algorithm, as shown by Kaplanyan and Dachsbacher [113].

5.4 TRANSIENT PATH INTEGRAL FRAMEWORK

We first extend the standard path integral formulation to transient state. This will allow us to formalize the notion of transient rendering, understand how to elevate steady state rendering to transient state, and, most importantly, identify the unique challenges of solving this more difficult light transport problem.

For transient rendering, in addition to integrating over spatial coordinates, we must also integrate over the space of temporal delays ΔT of all paths:

$$I = \int_{\Omega} \int_{\Delta T} f(\overline{\mathbf{x}}, \overline{\Delta \mathbf{t}}) \, \mathrm{d}\mu(\overline{\Delta \mathbf{t}}) \, \mathrm{d}\mu(\overline{\mathbf{x}}), \tag{5.11}$$

where $\overline{\Delta \mathbf{t}} = \Delta t_0 \dots \Delta t_k$ defines a sequence of time delays and $\mathrm{d}\mu(\overline{\Delta \mathbf{t}})$ denotes temporal integration at each path vertex.

We define the path contribution function $f(\overline{\mathbf{x}}, \overline{\Delta \mathbf{t}})$ as the original, but with the emission L_e, path throughput $\mathfrak{T}$, and sensor importance W_e additionally depending on time:

$$f(\overline{\mathbf{x}}, \overline{\Delta \mathbf{t}}) = L_e(\mathbf{x}_0 \to \mathbf{x}_1, \Delta t_0) \, \mathfrak{T}(\overline{\mathbf{x}}, \overline{\Delta \mathbf{t}}) \, W_e(\mathbf{x}_{k-1} \to \mathbf{x}_k, \Delta t_k). \tag{5.12}$$

The temporal sensor importance W_e now defines not only the spatial and angular sensitivity, but also the region of time we are interested in evaluating. This could specify a delta function at a desired time, or more commonly,
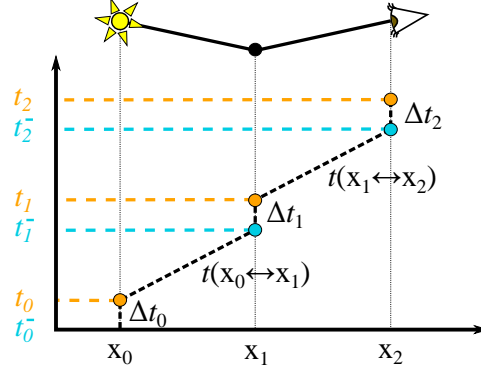
Figure 5.3: Spatio-temporal diagram of light propagation for a path with $k = 2$. Light is emitted at time t_0, and reaches $\mathbf{x}_1$ at $t_0 + t(\mathbf{x}_0 \leftrightarrow \mathbf{x}_1)$. After a microscopic temporal delay Δt_1, light emerges from $\mathbf{x}_1$ at t_1 and takes $t(\mathbf{x}_1 \leftrightarrow \mathbf{x}_2)$ time to reach $\mathbf{x}_2$. The sensor may include a further temporal delay Δt_2.

a finite interval of interest in the temporal domain (analogous to the shutter interval in steady state rendering, though at much smaller time scales). Likewise, the time parameter of the emission function L_e can define temporal variation in emission (e.g. pulses). The transient path throughput is now defined as:

$$\mathfrak{T}(\overline{\mathbf{x}}, \overline{\Delta \mathbf{t}}) = \left[\prod_{i=1}^{k-1} \rho(\mathbf{x}_i, \Delta t_i) \right] \left[\prod_{i=0}^{k-1} G(\mathbf{x}_i, \mathbf{x}_{i+1}) V(\mathbf{x}_i, \mathbf{x}_{i+1}) \right]. \tag{5.13}$$

Since we assume that the geometry is stationary (relative to the speed of light), the geometry and visibility terms depend only on the spatial coordinates of the path, as in steady state rendering. However, we extend the scattering kernel ρ with a temporal delay parameter Δt_i to account for potential time delays at each scattering vertex $\mathbf{x}_i$. Such delays can occur due to e.g. multiple internal reflections within micro-geometry [240], electromagnetic phase shifts in the Fresnel equations [55, 193], or inelastic scattering effects such as fluorescence [243, 64].

TIME DELAYS. A transient light path is defined in terms of spatial and temporal coordinates. The temporal coordinates at each path vertex $\mathbf{x}_i$ are t_i^-, the time immediately before the scattering event, and t_i, the time immediately after (see Figure 5.3). Both time coordinates can be obtained by accounting for all *propagation* delays between vertices $t(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i+1})$ and *scattering* delays Δt_i at vertices along the path:

$$t_i^- = \sum_{j=0}^{i-1} \left(t(\mathbf{x}_j \leftrightarrow \mathbf{x}_{j+1}) + \Delta t_j \right), \quad t_i = t_i^- + \Delta t_i, \tag{5.14}$$

where t_0 and t_k denote the emission and detection times of a light path. The transient simulation is assumed to start at $t_0^- = 0$. In the general case of non-linear media [62, 91, 3], propagation time along a path segment is:

$$t(\mathbf{x}_j \leftrightarrow \mathbf{x}_{j+1}) = \int_{s_j}^{s_{j+1}} \frac{\eta(\mathbf{x}_r)}{c} \, dr, \tag{5.15}$$

where r parametrizes the path of light between the two points, s_j and s_{j+1} are the parameters of the path at $\mathbf{x}_j$ and $\mathbf{x}_{j+1}$, respectively, c is the speed of light in vacuum and $\eta(\mathbf{x}_r)$ represents the index of refraction of the medium
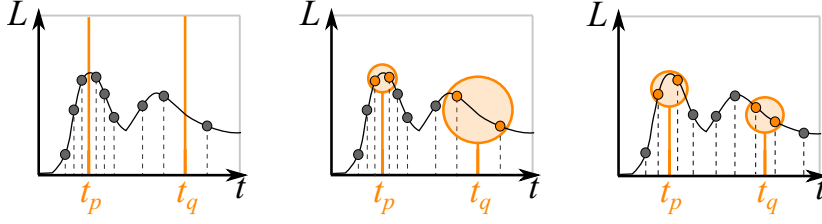
Figure 5.4: **Left:** The probability of finding a sample at a specific time instant (t_p or t_q) is nearly zero (Section 5.4). **Middle:** Density estimation on the temporal domain (Section 5.5) allows us to reconstruct radiance at any instant, although with varying bias and variance in time. **Right:** A more uniform distribution of samples in the temporal domain leads to more uniform bias and better reconstructions (Section 5.6).

at $\mathbf{x}_r$. In the typical scenario where η is constant along a path segment, Equation (5.15) reduces to a simple multiplication: $t(\mathbf{x}_j \leftrightarrow \mathbf{x}_{j+1}) = \|\mathbf{x}_j - \mathbf{x}_{j+1}\| \eta / c$. Figure 5.3 illustrates both the spatial and temporal dimensions of a path for the case of $k = 2$.

NUMERICAL INTEGRATION.   Similar to its steady state counterpart (5.1), the transient path integral (5.11) can be numerically approximated using a Monte Carlo estimator:

$$\langle I \rangle = \frac{1}{n} \sum_{j=1}^{n} \frac{f(\overline{\mathbf{x}}_j, \overline{\Delta \mathbf{t}}_j)}{p(\overline{\mathbf{x}}_j, \overline{\Delta \mathbf{t}}_j)}, \tag{5.16}$$

which averages n random paths $\overline{\mathbf{x}}_j, \overline{\Delta \mathbf{t}}_j$ drawn from a spatio-temporal probability distribution (pdf) $p(\overline{\mathbf{x}}_i, \overline{\Delta \mathbf{t}}_i)$ defined by the chosen path and time delay sampling strategy. In steady state, the pdf only needs to deal with the location of path vertices $\overline{\mathbf{x}}_i$.

5.4.1 *Challenges of sampling in transient state*

Equation (5.11) shows a new domain of scattering delays ΔT that must be sampled. Most existing path sampling techniques generate random paths incrementally, vertex-by-vertex, by locally importance sampling the scattering function ρ at each bounce, and optionally making deterministic shadow connections between light and camera subpaths. We could in principle elevate any such algorithm to transient state by simply sampling the transient scattering function $\rho(\mathbf{x}_i, \Delta t_i)$, instead of the steady state scattering function $\rho(\mathbf{x}_i)$.

Unfortunately, transient rendering poses hidden challenges, since *propagation* delays between vertices $t(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i+1})$ are fundamentally different than *scattering* delays Δt_i defined at the light, sensor, and interior vertices. While scattering delays reside on a separate sampling domain ΔT, propagation delays are a direct consequence of the spatial positions of path vertices sampled from Ω. Hence, if spatial positions are determined by a steady state sampling routine ignorant of propagation delays, control of the propagation time in a path's total duration t_k is lost, leaving only the scattering delays Δt_i to control t_k.

Other factors resulting from the temporal structure of light transport make any naïve extension to transient rendering extremely inefficient: to visualize transient effects, the time window of both the sensor and the light source needs to be small (≈ 10 *pico*seconds); moreover, scattering events re-

sult in *femto*second temporal delays. The temporal domain of the path contribution thus becomes a near delta manifold (i.e. a *caustic* in time), which is virtually impossible to sample by random chance. Since the total path duration t_k cannot be directly controlled, deterministic shadow connections are rendered useless, having little chance of finding a non-zero contribution in both the light L_e and the sensor W_e. In general, the probability of randomly finding non-zero contribution for a specific time decreases as either Δt_i, L_e or W_e get closer to delta functions in the temporal domain, which are precisely the cases of interest in transient light transport.

When several distinct measurements of the path integral have to be computed, a common optimization strategy is to share randomly sampled paths to estimate all measurements simultaneously. This technique (path reuse) is utilized in the spatial domain in light tracing and bidirectional path tracing to estimate all pixels in the image plane at once. A similar situation occurs in the transient domain, where each frame f defines a specific sensor importance function $W_e^f(\mathbf{x}_{k-1} \to \mathbf{x}_k, t_k)$ and the time window covered by all frames is significantly larger than the per-frame time window. We could therefore leverage temporal path reuse to improve the efficiency of steady state path sampling methods when applied to rendering transient light transport. In practice, for every generated random path in Equation (5.16), we could evaluate the contribution functions for every frame f, which differ only in the temporal window of the sensor importance function W_e^f.

This path reuse technique is equivalent to histogram density estimation [203] in the *temporal domain* of the sensor, where each bin of the histogram represents one frame, and the bin's width h is the frame duration. Unfortunately, this type of density estimation produces very noisy results, especially for bins with very small width (i.e. exposure time). This results in a low convergence rate of $O(n^{-1/3})$ [196], where n is the number of samples. This is illustrated in Figure 5.5: although obviously better than not reusing paths, results are still extremely noise even with a large amount of samples. Still, this suggests that more elaborated density estimation techniques may lead to better convergence rates and/or less noisy reconstructions.

In the following, we first show how kernel-based density estimation techniques in the temporal domain allow us to reconstruct radiance along time from a sparse set of samples (see Section 5.5 and Figure 5.4, middle). Then, we show how a skewed temporal sample distribution affects radiance reconstruction, and develop a set of sampling strategies for participating media that enable some control over propagation delays, leading to a more uniform distribution of samples in time and therefore more accuracy (see Section 5.6 and Figure 5.4, right).

5.5 KERNEL-BASED TEMPORAL DENSITY ESTIMATION

Kernel-based density estimation is a widely known statistical tool to reconstruct a signal from randomly sampled values. These techniques significantly outperform histogram-based techniques (like the path reuse described above), especially for noisy data [203]. A kernel with finite bandwidth is used to obtain an estimate of the value of a signal at a given point by computing a weighted average of the set of random samples around such point. We thus introduce a temporal kernel $K_{\mathcal{T}}$ with bandwidth $\mathcal{T}$ to esti-
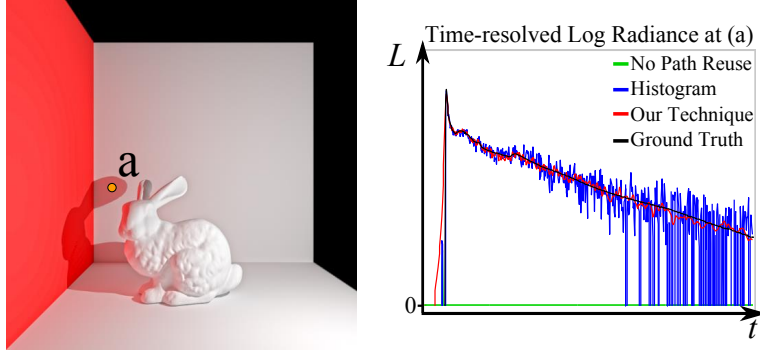
Figure 5.5: Time-resolved irradiance computed at pixel (a) in the scene on the left using, no path reuse (green), histogram-based path reuse (red), and kernel-based path reuse (blue), for the same number of samples. Without path reuse it is extremely difficult to reconstruct the radiance, since the probability of finding a path arriving at the specific frame is close to zero. This is solved using path reuse, although with different levels of improvement: while histogram-based density estimation shows a very noisy result, our proposed progressive kernel-based estimation shows a solution with significantly lower variance, while preserving high-frequency features due to the progressive approach.

mate incoming radiance I at the sensor at time t as a function of n samples of I:

$$\langle I_n \rangle = \frac{1}{n} \sum_{j=1}^{n} K_{\mathcal{T}}(\|t - t_{k,j}\|)\widehat{I}_j, \tag{5.17}$$

where $\widehat{I}_j = f(\overline{\mathbf{x}}_j, \overline{\Delta \mathbf{t}}_j)/p(\overline{\mathbf{x}}_j, \overline{\Delta \mathbf{t}}_j)$ is the contribution of path $\overline{\mathbf{x}}_j$ in the measured pixel, and $t_{k,j}$ is the total time of the path (5.14). Using this temporal density estimation kernel reduces variance, but at the cost of introducing bias (see Figure 5.4, middle). This can be solved by using consistent progressive approximations [68, 121], which converge to the correct solution in the limit.

Inspired by these works, we model our progressive density estimation along the temporal domain, for which we rely on the probabilistic approach for progressive photon mapping used by Knaus and Zwicker [121]. We compute the estimate $\langle I_n \rangle$ in n steps, progressively reducing bias while allowing variance to increase; this is done by reducing the kernel bandwidth $\mathcal{T}$ in each iteration as $\mathcal{T}_{j+1}/\mathcal{T}_j = (j + \alpha)/(j + 1)$. The variance of our temporal progressive estimator vanishes with $O(n^{-\alpha})$ as expected, since the shrinking ratio is inversely proportional to the variance increase factor. Bias, on the other hand, vanishes with $O(n^{-2(1-\alpha)})$. Note that the parameter α defines the convergence of *both* sources of error (bias and variance). To find the optimal value that minimizes both, we use the *asymptotic mean square error* (AMSE), defined as:

$$\text{AMSE}(\langle I_n \rangle) = \text{Var}[I_n] + \text{E}[\epsilon_n]^2. \tag{5.18}$$

Using the convergence rate for both bias and variance, we find that the optimal α that minimizes the AMSE is $\alpha = 4/5$, which leads to a convergence of $O(n^{-4/5})$. This is significantly faster than using the histogram method, $O(n^{-1/3})$, which we illustrate in Figure 5.5. The detailed derivation of the behavior of the algorithm can be found in Section 5.A.
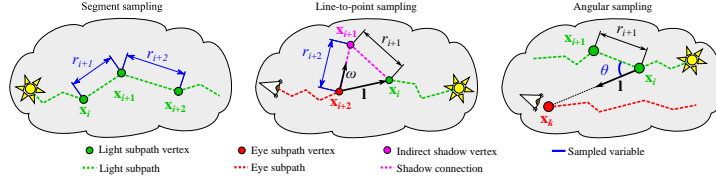
Figure 5.6: Sampling strategies for participating media with a uniform distribution in the time domain. **Left:** Sampling scattering distance for a light subpath. This strategy can also be applied to eye subpaths. **Middle** Sampling shadow connections through a new indirect vertex: line-to-point sampling of shadow connections. **Right:** Sampling the angular scattering function (phase function).

5.5.1 *Transient progressive photon mapping.*

Our approach above is agnostic to the algorithm used to obtain the samples (e.g. samples in Figure 5.5 have been computed using path tracing). This means that it can be combined with biased density estimation-based algorithms such as (progressive) photon mapping [109, 68, 67], which is well suited for complex light paths such as spatial caustics. However, although using progressive photon mapping as the source of samples for our temporal density estimation is consistent in the limit, it results in suboptimal convergence due to the coupling of the bias and variance between the spatial and temporal kernels. Instead, we introduce the temporal domain into the photon mapping framework, by adding the temporal smoothing kernel $K_{\mathcal{T}}$ in the radiance estimation [16]. Radiance $\widehat{L_o}(\mathbf{x}, t)$ is estimated using M photons with contribution γ_i as

$$\widehat{L_o}(\mathbf{x}, t) = \frac{1}{M} \sum_{i=1}^{M} K(\|\mathbf{x} - \mathbf{x}_i\|, \|t - t_i^-\|)\gamma_i. \tag{5.19}$$

Combining both kernels into a single multivariate kernel allows controlling the variance increment in each step as a function of a single α, so that it increments at a rate of $(j+1)/(j+\alpha)$, while reducing bias by progressively shrinking both the spatial and temporal kernel bandwidths (R and $\mathcal{T}$ respectively). As shown in Section 5.B, these are reduced at each iteration j following:

$$\frac{\mathcal{T}_{j+1}}{\mathcal{T}_j} = \left(\frac{j+\alpha}{j+1}\right)^{\beta_{\mathcal{T}}}, \qquad\qquad \frac{R_{j+1}^2}{R_j^2} = \left(\frac{j+\alpha}{j+1}\right)^{\beta_R}, \tag{5.20}$$

where $\beta_{\mathcal{T}}$ and β_R are scalars in the range $[0, 1]$ controlling how much each term is to be scaled separately, with $\beta_{\mathcal{T}} + \beta_R = 1$. The convergence rate of the *combined* spatio-temporal density estimation is $O(n^{-4/7})$[1]. Using this formulation allows us to handle complex light paths in transient state, while still progressively reducing bias and variance introduced by both progressive photon mapping and our temporal density estimation, in the spatial and temporal domains respectively. We refer to Section 5.B for the detailed description of the algorithm, including the full derivation of the error and convergence rate.

---

1 Note that a naïve combination of the temporal (1D) and the spatial (2D) kernels would yield a slower convergence than the combined 3D kernel convergence $O(n^{-4/7})$ when using the optimal parameters $\alpha = 4/7$ and $\beta_{\mathcal{T}} = 1/3$ reported in previous work [113] (for volumetric density estimation) or in the statistics literature [196].

5.6 TIME SAMPLING IN PARTICIPATING MEDIA

As we mentioned earlier (Section 5.4.1), the performance of our transient density estimation techniques can be further improved by a more uniform distribution of samples in time. This makes the relative error uniform in time and optimizes convergence (see Figure 5.4, right). Steady state sampling strategies aim to approximate radiance (path contribution). Since more radiant samples happen at earlier times (due to light attenuation), these sampling techniques skew the number of samples towards earlier times. As a consequence, there is a increase of error along time (see Figures 5.8 and 5.10). New sampling strategies are therefore needed for transient rendering.

Sampling strategies over scattering delays Δt_i have a negligible influence over the total path duration t_k (Figure 5.3). For surface rendering, scattering delays are the only control that sampling strategies can have on the temporal distribution of samples, and there is therefore little control over the total path duration. In participating media, however, sample points can be potentially located anywhere along the path of light, providing direct control also over the propagation times $t(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i+1})$. In this section we develop new sampling strategies for participating media that target a uniform sample distribution in the time domain, by customizing:

- The pdf for each segment of the camera or light subpath (Section 5.6.1).
- The pdf for a shadow connection (connecting a vertex of the camera path to a vertex of the light path) via an additional vertex (Section 5.6.2).
- The pdf in the angular domain to obtain the direction towards the next interaction (Section 5.6.3).

Each of these sampling strategies ensures a uniform distribution of samples in time for each particular domain of the full path. Although this does not statistically ensure uniformity for the whole path, in practice the resulting distribution of total path duration t_k samples in time is close to uniform and therefore noise is reduced (the improvement over steady state strategies is discussed in Section 5.7). Note that these strategies are also agnostic of the properties of the media (except for the index of refraction), and can therefore be used in arbitrary participating media. Additionally, they can be combined with steady-state radiance sampling via multiple importance sampling (MIS) [220].

5.6.1 *Sampling scattering distance in eye/light subpaths*

Each of the segments of a subpath in participating media often shares the same steady-state sampling strategy, such as mean-free-path sampling, which does not necessarily ensure a uniform distribution of temporal location of vertices. We aim to find a pdf $p(r)$ (where r is the scattering distance along one of the subpath segments) so that the probability distribution $p\left(\cup_{i=1}^{\infty} t_i\right)$ of temporal subpath vertex locations is uniform (see Figure 5.6, left). We first define $p\left(\cup_{i=1}^{\infty} t_i\right)$ based on the combined probability distribution $p(t_i)$ (temporal location of vertex $\mathbf{x}_i$ in the light subpath) for all subpath vertices:

$$p\left(\cup_{i=1}^{\infty} t_i\right) = \sum_{i=1}^{\infty} p(t_i), \tag{5.21}$$

where $p(t_i)$ is recursively defined based on $p(t_{i-1})$. Given that $t_i = t(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i-1}) + t_{i-1}$, as shown in Equation (5.14), we have

$$p(t_i) = \int_0^{t_i} p\big(t(\mathbf{x}_{i-1} \leftrightarrow \mathbf{x}_i)\big) p(t_{i-1}) dt_{i-1}, \tag{5.22}$$

$$p(t_1) = p\big(t(\mathbf{x}_0 \leftrightarrow \mathbf{x}_1)\big), \tag{5.23}$$

since the probability of the addition of two random variables is the convolution of their probability distributions. $p\left(t(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i-1})\right)$ is the probability distribution of the propagation time, which is related to the scattering distance pdf $p(r)$ by a simple change of variable $r = \frac{c}{\eta} t(\mathbf{x}_{i-1} \leftrightarrow \mathbf{x}_i)$. Note that, in this notation, we are assuming (as previously discussed) that scattering delays Δt_i are negligible compared to propagation time. This definition is analogous for the eye subpath.

We show in Section 5.C.1 that the exponential distribution $p(r) = \lambda e^{-\lambda r}$ ensures that $p\left(\cup_{i=1}^{\infty} t_i\right)$ follows a uniform distribution for any λ parameter. Figure 5.7 (left) experimentally shows that this exponential distribution leads to this uniform probability for the whole subpath, while a uniform pdf leads to a non-uniform temporal sample distribution. In practice, λ modulates the average number of segments of the subpath: for a path ending at time t_e, the average number of segments with path duration $t_k \leq t_e$ is $\lambda \frac{c}{\eta} t_e$. Our results show that an average of three or four vertices per subpath gives a good compromise between path length, efficiency and lack of correlation. Note that mean-free-path sampling is also an exponential distribution whose rate equals the extinction coefficient of the medium ($\lambda = \sigma_t$). Directly using mean-free-path sampling is thus optimal for time sampling when σ_t is close to the optimal λ.

SUBPATH TERMINATION.   Russian roulette is a common strategy in steady state rendering algorithms. It probabilistically terminates subpaths at each scattering interaction, reducing longer paths with a small radiance contribution. In transient state, this unfortunately translates into fewer samples as time advances, reducing the signal-to-noise ratio (SNR) at higher frames. Instead, we simply terminate paths with a total duration greater than the established time frame.

While the temporal locations of subpath vertices are uniform, there is still little control over the spatial locations $\mathbf{x}_i$. These depend not only on scattering distances but also on scattering angles. As shadow rays are deterministic and depend on such spatial locations, uniformity cannot be ensured. To address this, we develop a new strategy that deals with such shadow connections (Section 5.6.2) and an angular sampling strategy (Section 5.6.3) that leads to an improved distribution in the temporal domain of the location-dependent propagation delays.

5.6.2   *Sampling line-to-point shadow connections*

Shadow rays are deterministic segments connecting a vertex in the eye subpath to another vertex in the light subpath, so their duration cannot be controlled. We introduce a new *indirect* shadow vertex whose position can be stochastically set to ensure a uniform sample distribution along the duration of the (extended) shadow connection. The geometry of this indirect connection is similar to equiangular sampling [112, 187, 130] (see Figure 5.6, middle).

Given a vertex $\mathbf{x}_i$ of a light subpath, a vertex $\mathbf{x}_{i+2}$ and a direction ω (importance sampled from the scattering function) on an eye subpath, our technique connects the two vertices via an indirect bounce at an importance-sampled location $\mathbf{x}_{i+1}$. If r_{i+1} and r_{i+2} are the distances from $\mathbf{x}_{i+1}$ to $\mathbf{x}_i$ and $\mathbf{x}_{i+2}$ respectively, we importance sample r_{i+2} to enforce a uniform propagation time between the connected vertices $\{\mathbf{x}_i, \mathbf{x}_{i+1}, \mathbf{x}_{i+2}\}$. This connection could also be done in reverse order (from $\mathbf{x}_{i+2}$ to $\mathbf{x}_i$).

Given $\mathbf{l} = \mathbf{x}_i - \mathbf{x}_{i+2}$ and a connection time range (t_a, t_b) (in which we aim to get uniformly distributed samples), the pdf is:

$$p(r_{i+2}) = \frac{\eta}{c(t_b - t_a)} \left[1 + \frac{r_{i+2} - (\mathbf{l} \cdot \omega)}{\sqrt{r_{i+2}^2 - 2r_{i+2}(\mathbf{l} \cdot \omega) + (\mathbf{l} \cdot \mathbf{l})}} \right], \tag{5.24}$$

which leads to the following inverse cumulative distribution function (cdf):

$$r_{i+2}(\xi) = \frac{(\xi(t_b - t_a) + t_a - t_i - \Delta t_{i+1})^2 - \left(\frac{\eta}{c}\right)^2 (\mathbf{l} \cdot \mathbf{l})}{2\frac{\eta}{c}(\xi(t_b - t_a) + t_a - t_i - \Delta t_{i+1}) - 2\left(\frac{\eta}{c}\right)^2 (\mathbf{l} \cdot \omega)}. \tag{5.25}$$

where $\xi \in [0, 1)$ is a random number. Assuming a rendered temporal range of $(0, t_e)$, we set the shadow connection limits to $t_a = t_i + t(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i+2})$ and $t_b = t_e - \Delta t_k - \left(\sum_{j=i+2}^{k-1} t(\mathbf{x}_j \leftrightarrow \mathbf{x}_{j+1}) + \Delta t_j \right)$. The derivation of this pdf can be found in Section 5.C.2. Figure 5.7 (middle) compares our line-to-point sampling strategy with other common strategies in terms of sample distribution along the temporal domain, leading to a uniform distribution of samples. Note that we discard all paths with a total duration larger than t_e (when $t_b < t_a$).

5.6.3 *Angular sampling*

Importance sampling the phase function generally leads again to a suboptimal distribution of samples in time. We propose a new angular pdf $p(\theta)$ to be applied at each interaction of the light subpath, which targets the temporal distribution of samples assuming that the next vertex $\mathbf{x}_{i+1}$ casts a deterministic shadow ray towards the sensor. Given the sensor vertex $\mathbf{x}_k$ and a sampled distance r_{i+1} between two consecutive vertices $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$ (see Figure 5.6, right), this strategy ensures a uniform distribution of the total propagation time in $\{\mathbf{x}_i, \mathbf{x}_{i+1}, \mathbf{x}_k\}$. The direction from $\mathbf{x}_i$ to $\mathbf{x}_{i+1}$ is $\omega = (\theta, \phi)$ (in spherical coordinates) where θ is the sampled angle and ϕ is uniformly sampled in $[0..2\pi)$. Note that the sampled angle θ is related to the direction towards the sensor ($\mathbf{l} = \mathbf{x}_k - \mathbf{x}_i$) and not to the incoming direction (which is often the system of reference for phase function importance sampling). This pdf is:

$$p(\theta) = \frac{r_{i+1} \sin \theta}{2\sqrt{r_{i+1}^2 + |\mathbf{l}|^2 - 2r_{i+1}|\mathbf{l}| \cos \theta}}, \tag{5.26}$$

with the following inverse cdf:

$$\theta(\xi) = \arccos \left(\frac{|\mathbf{l}| - 2r_{i+1}^2 \xi^2 - 2\xi r_{i+1} (|\mathbf{l}| - 1)}{r_{i+1}|\mathbf{l}|} \right). \tag{5.27}$$

Section 5.C.3 contains the full derivation. This pdf prioritizes segments towards the target vertex $\mathbf{x}_t$, which helps in practice since backward directions often lead to paths that become too long for the rendered time frame.
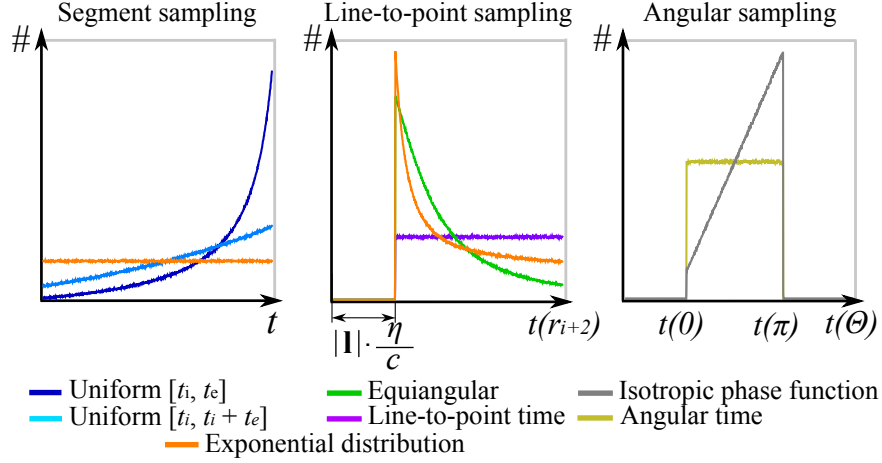
Figure 5.7: Histogram of the number of samples along the temporal dimension for different sampling strategies. **Left:** Sample distribution for the whole light subpath, according to the importance sampling of subpath segments. **Middle:** Importance sampling of a line-to-point shadow connection **Right:** Angular importance sampling. Notice how our developed sampling strategies (exponential for segment sampling and the corresponding time sampling strategies in the other two cases) lead to a uniform distribution of samples along the temporal domain on each case. Both the line-to-point and the angular sampling are defined over a certain range.

Figure 5.7 (right) shows how our angular sampling strategy leads to a uniform distribution of samples in time, as opposed to other alternatives. The shadow ray from vertex $\mathbf{x}_{i+1}$ to the sensor in $\mathbf{x}_k$ (and to every vertex in the eye subpath in bidirectional path tracing) is then cast by applying the sampling technique described in Section 5.6.2. Alternatively, the shadow ray could be cast from $\mathbf{x}_i$ by applying MIS between this angular sampling and line-to-point time sampling (Section 5.6.2). We also apply the same angular sampling strategy for each interaction of the eye subpath, targeting the light source.

5.7 RESULTS

Here we show and discuss our rendered scenes. For visualization we use selected frames of the animations; for the complete animations, we refer the reader to the supplementary video[2].

In all the scenes light emission occurs at $t = 0$ with a delta pulse[3]. Unless otherwise stated, we use transient path tracing and kernel-based density estimation (Section 5.5) for sampling and reconstruction, respectively. For the latter, we use a Perlin [175] smoothing kernel, following previous work [66, 113], with forty nearest neighbors to determine the initial kernel bandwidth. Unless noted otherwise, all results are shown in camera time [224] (i.e. including the propagation time of the last segment).

---

2 http://giga.cps.unizar.es/~ajarabo/pubs/transientSIGA14/videos/Jarabo2014_main_video.mp4

3 We could use a Gaussian pulse, although this would introduce a number of downsides: 1) an ideal delta pulse does not introduce any additional temporal blur; 2) in reality, the scale of physical Gaussian pulses is 2-3 orders of magnitude smaller than the shutter open interval, in effect constituting a delta pulse; and 3) a delta pulse allows us to distinguish between effects caused by the actual behavior of light and effects due to limitations of current hardware.
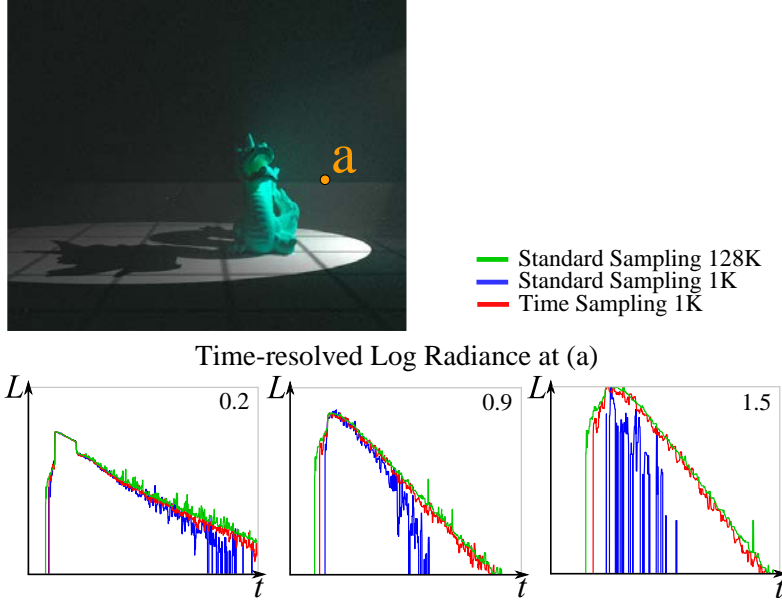
Figure 5.8: Comparison of our three time sampling strategies combined, against the standard techniques used in steady state, in the *dragon* scene accounting for multiple scattering (top). Each graph shows the time-resolved radiance (bottom) at pixel (a), for three different scattering coefficients $\sigma_s = \{0.2, 0.9, 1.5\}$, and absorption $\sigma_a = 0.1$. For 1K samples per pixel and frame, our combined techniques (red) feature a similar quality as standard steady state techniques with 128 times more samples (green), while with the same number of samples, our techniques significantly outperform standard sampling (blue), especially in highly scattering media. To emphasize the differences between sampling techniques, here we use the histogram path reuse (see Section 5.5). Additional results for other types of media can be found in Figure 5.9.

Figure 5.8 and Figure 5.9 compares transient rendering using our three time-based sampling strategies (Section 5.6) against common radiance-based steady-state sampling techniques (mean-free-path and phase-function sampling, and deterministic shadow connection) for isotropic and anisotropic media, respectively. Our approach distributes samples more uniformly in time, which reduces variance along the whole animation, while significantly lowering noise in later frames. We obtain similar quality to standard sampling using two orders of magnitude less samples. These advantages are even more explicit when using our line-to-point sampling strategy to render single scattering, as shown in Figure 5.10, where we compare against equiangular sampling [130]. Figure 5.11 shows how the *combination* of our kernel-based density estimation and our time sampling strategies produces better results than using either technique in isolation.

Figures 5.1, 5.13 and 5.12 demonstrate the macroscopic delays due to traversing media with different orders of refraction, which leads to a temporal delay of the wavefront, especially visible in the caustics. In these examples, we use a transient version of the *photon beams* algorithm [106] to obtain the radiance samples due to scattering in the media. For the single image visualization of Figure 5.1 we use the *peak-time* visualization proposed by Velten et al. [224]. Another example of macroscopic time delay can be observed in the mirror ball in Figure 5.12 (bottom), where the refracted image shows a delay with respect to the actual events in the world, due to the additional path length introduced by reflection. In both scenes of this figure
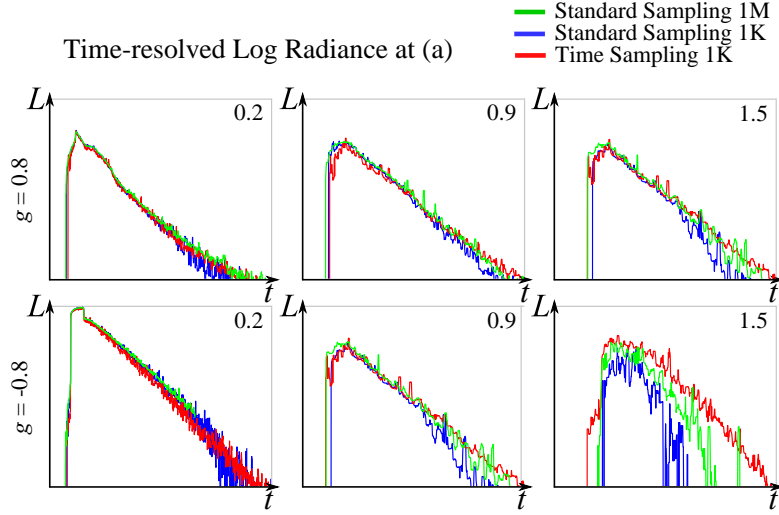
Figure 5.9: Comparison of our three time sampling strategies combined, against the standard techniques used in steady state, in the *dragon* scene of Figure 5.8, for three different scattering coefficients $\sigma_s = \{0.2, 0.9, 1.5\}$, and absorption $\sigma_a = 0.1$, for anisotropic media with $g = 0.8$ and $g = -.08$. For 1K samples per pixel and frame, our combined techniques (red) feature a similar quality as standard steady state techniques with 1000 times more samples (green), while with the same number of samples, our techniques significantly outperform standard sampling (blue), especially in highly scattering media. This shows that, as the directionality of the scattering in the media increases, our technique results in higher performance than standard techniques (note that in Figure 5.8 the results with similar quality for standard sampling are computed for 128K samples, while in this case we compare against 1M samples). As in Figure 5.8, we use the histogram path reuse for reconstruction (see Section 5.5).

we can also see the different orders of scattering in the surfaces, which gets diffuser as time advances.

The temporal delay due to refraction becomes particularly interesting simulating real-world glass (Figure 5.14): This scene, computed using the transient progressive photon mapping described in Section 5.5 that allows us to robustly render complex paths such as caustics, shows how as light traverses the glass, its wavelength-dependent index of refraction causes chromatic dispersion even when light incomes perpendicular to the surface normal. This could be used to obtain the wavelength-dependent index of refraction of different crystals, or the power spectra of a light source by using a perpendicular incident beam. Moreover, this refraction delay is different between ordinary and extraordinary rays in birefringent crystals [232, 134] (Figure 5.15). Figure 5.16 shows an example of the effect of the delay produced at scattering events $\rho_s(\mathbf{x}_{i-1} \to \mathbf{x}_i \to \mathbf{x}_{i+1}, \Delta t_i)$, for the particular case of fluorescence. The object's material is chlorophyll, which re-emits energy at 680 nm after absorption [64].

Finally, Figure 5.17 compares our simulation against a real scene captured with the femtophotography technique of Velten et al. [224]. We can see that our simulation faithfully reproduces the different orders of scattering events occurring during light propagation.
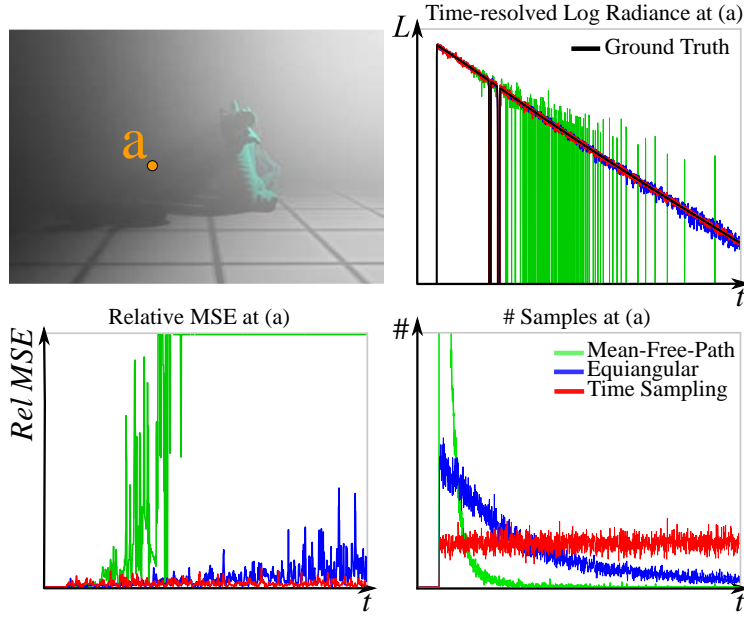
Figure 5.10: Comparison of different sampling techniques for computing single scattering, in a scene consisting of a dragon illuminated by point light source within participating media (left). As opposed to simple *mean-free-path sampling* and the state-of-the-art *equiangular sampling* [130], that distributes samples based on radiance, our point-to-line sampling (Section 5.6.2) distributes samples so that the are uniformly distributed in time (bottom, right). This allows performing better in terms of relative error (bottom, left) when rendering time-resolved radiance, avoiding the radiance signal degradation at longer times. Here we use the histogram (Section 5.5) to emphasize the performance of the algorithms.

5.8 DISCUSSION

In summary, we have extended the classical path space integral to include the temporal domain, and shown how the high frequency nature of transient light transport leads to severe sampling problems. We have proposed novel sampling strategies and density estimation techniques, which allow us to distribute samples uniformly in time, resulting in reduced variance and a constant distribution of bias. Sections 5.A, 5.B and 5.C contain a rigorous mathematical analysis of all our technical contributions. Last, we have presented simulations of interesting transient light transport effects using modified versions of a representative cross section of common rendering algorithms.

Apart from educational benefits, our work could be used to help design prototypes of novel ultra-fast imaging systems, or as a forward model for inverse problems such as recovering hidden geometry or material estimation. Our temporal progressive density estimation (Section 5.5) could also be used to accelerate radiance reconstruction in time-resolved imaging techniques, reducing the need for taking repeated measurements to improve the SNR. Moreover, synthetic ground truth data may become a very valuable tool for designing and benchmarking future ultra-fast imaging devices.

Our time-resolved simulations can help analyze the complex phenomena involved in light transport, and gain new insights. For instance, Figure 5.18 shows how during the early stages of light propagation, the first orders of scattering determine the shape of the light distribution (a spherical wave-
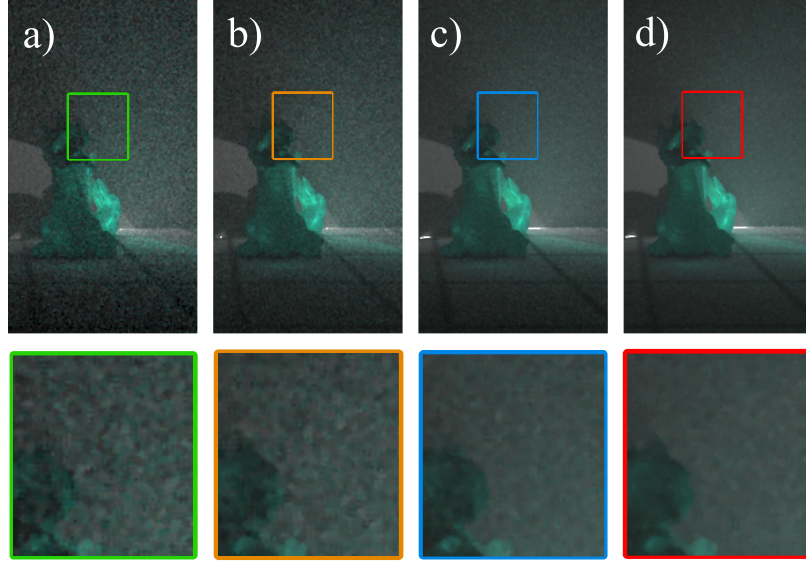
Figure 5.11: Selected frame of the *dragon* scene with $\sigma_s = 0.2$, rendered with a) standard sampling and histogram, b) our time sampling and histogram, c) standard sampling and our kernel-based density estimation, and d) time sampling and kernel-based density estimation. We can see how using our techniques combined lead to frames with significantly lower noise.

front), but over time this shape becomes a Gaussian of increasing variance. This observation is consistent with previous work [253], where it is shown that light in a medium exhibits diffusion after traveling about ten times the mean-free-path, and might explain some of the errors near the light source reported in the quantized diffusion model [29]. This effect is more accentuated in the presence of anisotropic media, where the wavefront behavior is even more dominant.

FUTURE WORK.   There are many compelling avenues of future work: First, it would be interesting to extend a unified path sampling framework [129] to transient state. We have shown how the photon beams algorithm [107] can be used in transient rendering, combined with our temporal density estimation; however, a spatio-temporal progressive photon beams framework would be needed to achieve optimal convergence in transient state. Additionally, by building a joint sampling strategy in both angle and distance, as in recent advanced steady state sampling techniques [166, 50], we could leverage the benefits of both to ensure better uniformity in the temporal distribution of samples. Furthermore, the three proposed time sampling strategies are limited to participating media; extending this to surface transport results in a much narrower sampling space. Metropolis Light Transport techniques [221] represent promising candidates in this regard, where temporal mutation strategies would be needed.

We hope that our research will inspire future work on our understanding of light transport, the design of ultra-fast imaging and the development of novel rendering techniques. For instance, several geometric approaches to acoustic rendering are also based on ray tracing: a more extensive analysis of similarities between acoustic and transient rendering might prove fruitful
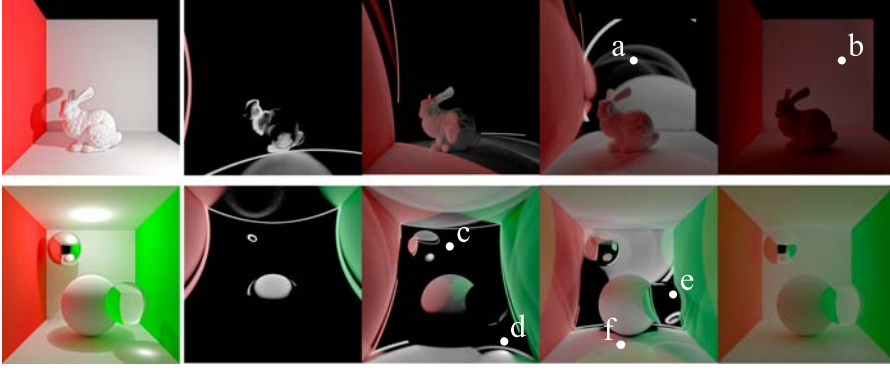
Figure 5.12: Light propagation in the absence of participating media, in the two scenes depicted in the leftmost column. In the *bunny* scene (top) we can see the curved shape of the indirect wavefronts, and how the irregular shape of the bunny model makes several different indirect wavefronts (a). In contrast the *spheres* scene we can observe the different delays due to longer optical paths in the mirror (c) and glass (e) spheres, and the primary (d) and secondary caustics (f). At longer times, when light has been reflected several times, the directionality is lost, and light is fundamentally diffuse (b).

to both domains. Our code and datasets (scenes and movies) are publicly available at the project page[4].

APPENDICES

5.A   PROGRESSIVE TEMPORAL DENSITY ESTIMATION

In this section we analyze the behavior and convergence of the error of the kernel-based temporal density estimation described in Section 5.5. We first analyze the variance and expected error introduced by using the density estimation kernel with a fixed bandwidth (Section 5.A.1). Based on these results, we then analyze the error and convergence rate of the progressive density estimation scheme, which allows to obtain a consistent estimation in the limit (Section 5.A.2). Finally, we derive the parameters yielding optimal convergence with respect to the AMSE for the progressive approach (Section 5.A.3).

5.A.1   *Variance and expected error of density estimation*

Following the recent probabilistic framework for the progressive photon mapping algorithm [121], here we analyze the variance and expected value of the error introduced by the temporal kernel-based density estimate at each iteration. This error ϵ is defined as the difference between the estimated pixel value $\langle I \rangle$ and the actual value I, at sensor point $\mathbf{x}$ and time t. Using the temporal kernel $K_{\mathcal{T}}$, with bandwidth $\mathcal{T}$, we have:

$$\epsilon = \frac{1}{n} \sum_{j=1}^{n} K_{\mathcal{T}}(\|t - t_j^-\|)\widehat{I}_j - I. \tag{5.28}$$
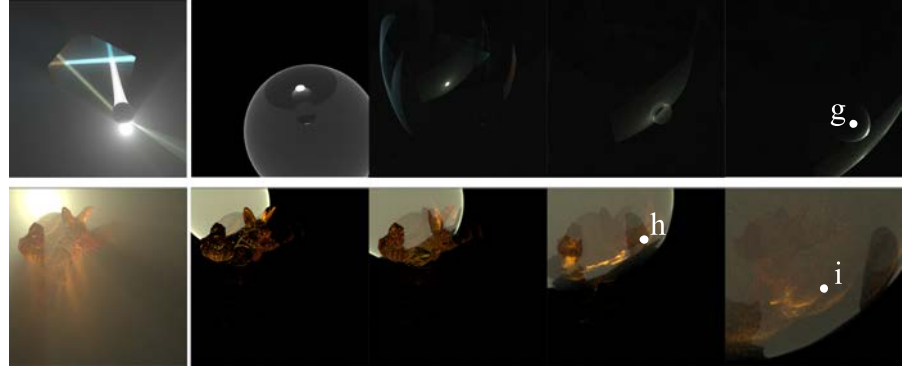
Figure 5.13: Selected frames from animations of transient light propagation in two scenes with scattering media (the steady state rendering for both is in the leftmost column) computed using *transient photon beams*. In both scenes we can see the delay produced in the caustics by the higher index-of-refraction of the crystal media (g,h), and how the different geometries generate different caustic patterns (g,i). We can also observe how the caustics are scattered and suffer extinction as they advance through the medium.
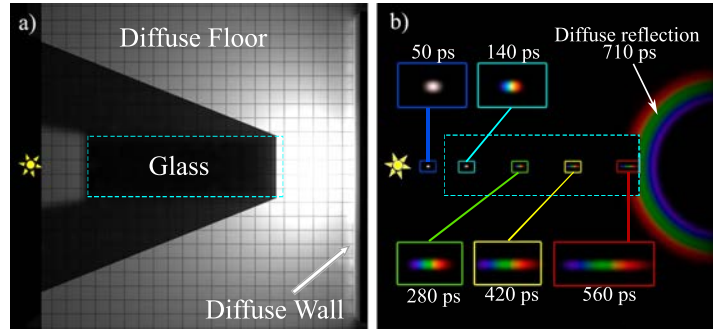
Figure 5.14: An orthogonal view of a scene with a white light pulse traversing perpendicularly a cube made of glass with index of refraction that varies linearly with the wavelength, in the range of [1.5,1.65], and hitting a wall after traversing the cube, transforming the interaction point into a virtual light source which illuminates a ground floor (b). Due to different speeds, the different wavelengths of the pulse are decomposed along the trajectory, even after the interaction. White light, as would be seen in steady state (a), therefore becomes a rainbow in the temporal domain.
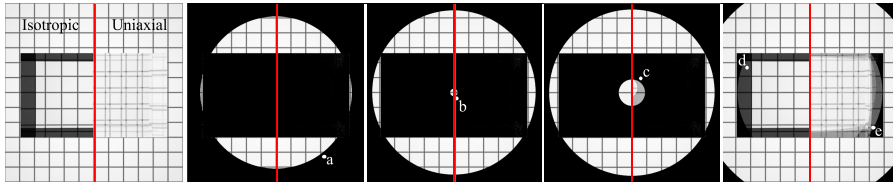
Figure 5.15: Light propagation in a piece of glass placed in front of a display with continuous emission in time; the left half of the image an isotropic crystal (with transmission governed by Snell's law) and the right half an uniaxial birefringent crystal. In both cases, the ordinary index of refraction is 1.5, and the extraordinary η_e in the birefringent is 1.65. First, direct light from the display arrives the camera (a); then, the refracted image begins to form (b). The extraordinary image appears instants later than the ordinary (c), due to higher index of refraction. Also note that ordinary refraction in the uniaxial crystal has lower energy than in the isotropic. Finally, internal reflections with longer optical paths are formed (d,e).

Figure 5.16: Fluorescent bunny illuminated by a pulse of light from a point light source. Light reaches the bunny, which reflects light centered in the green-ish spectra (b), and the rest is absorbed. After 10 ns, part of the absorbed light gets re-emitted at lower quantum energy, centered at 680 nm (c). Because of this re-emission, the hue of the bunny in steady state (a) is shifted towards yellow.
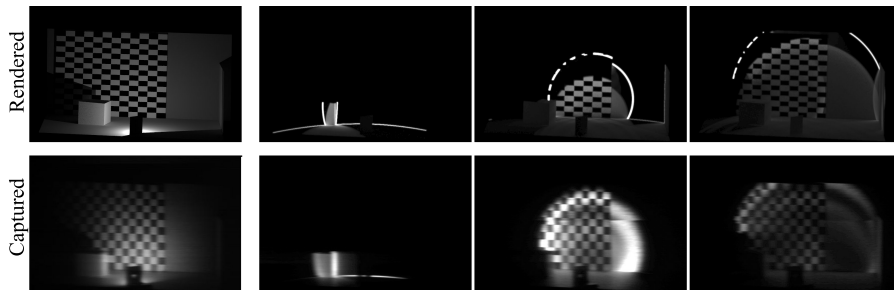
Figure 5.17: Comparison between the *Cube* scene from [224] and our rendered simulation of the same scene. Visible differences are due to approximate materials and camera properties.
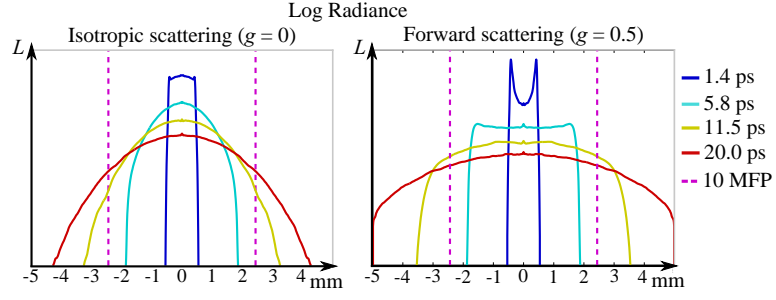
Figure 5.18: Time-resolved light transport from a point light source placed in the middle of an isotropic (left) and forward (right) scattering medium, emitting at time $t = 0$. Both media have a mean free path of 0.244 mm ($\sigma_t = 4.1 \text{ mm}^{-1}$). In the initial phase the light distribution is dominated by the wavefront shape of the low-order scattering events. In isotropic scattering, light distribution becomes Gaussian after traveling ten times the mean free path. In forward scattering, this distance is increased.

VARIANCE   In order to compute the variance of the error $\text{Var}[\epsilon]$ we need to make a set of assumptions: First, we assume that the samples' probability density is constant within the kernel $K_{\mathcal{T}}$ in the temporal domain. We denote this probability as $p_{\mathcal{T}}(t)$. We also assume that the samples' time t_j and pixel measurement $\widehat{I}_j$ are independent samples of the random variables T and $\widehat{I}$, respectively, where T has probability density $p_{\mathcal{T}}(t)$. Finally, we assume that the random variables T and $\widehat{I}$ are mutually independent. We thus model $\text{Var}[\epsilon]$ as:

$$
\begin{aligned}
\text{Var}[\epsilon] &= \text{Var}\left[\frac{1}{n}\sum_{j=1}^{n} K_{\mathcal{T}}(\|t - T\|)\widehat{I} - I\right] \\
&= \frac{1}{n}\left(\text{Var}[K_{\mathcal{T}}] + \text{E}[K_{\mathcal{T}}]^2\right) \\
&\quad \left(\text{Var}[\widehat{I}] + \text{E}[\widehat{I}]^2\right) - \frac{1}{M}\text{E}[K_{\mathcal{T}}]^2\text{E}[\widehat{I}]^2.
\end{aligned} \tag{5.29}
$$

Here $\text{E}[K_{\mathcal{T}}] = p_{\mathcal{T}}(t)$, while the variance introduced by the temporal kernel $\text{Var}[K_{\mathcal{T}}]$ has the form:

$$
\text{Var}[K_{\mathcal{T}}] = \int_{\Omega_{\mathcal{T}}} K_{\mathcal{T}}(\|t - T\|)^2 p_{\mathcal{T}}(T) \, dT - p_{\mathcal{T}}(t)^2, \tag{5.30}
$$

where $\Omega_{\mathcal{T}}$ is the area where $K_{\mathcal{T}}$ is compactly supported, with constant density of samples $p_{\mathcal{T}}(t)$. We express $K_{\mathcal{T}}$ as a canonical kernel $k_{\mathcal{T}}$ with unit integral such that $K_{\mathcal{T}}(\xi) = k_{\mathcal{T}}(\xi/\mathcal{T})\mathcal{T}^{-1}$, and perform the change of variable $\psi = (\xi - t)/\mathcal{T}$ and $d\xi = \mathcal{T} \, d\psi$:

$$
\int_{\Omega_{\mathcal{T}}} K_{\mathcal{T}}(\|\xi - t\|)^2 \, d\xi = \int_{\mathbb{R}} \frac{1}{\mathcal{T}} k_{\mathcal{T}}(\psi)^2 \, d\psi, \tag{5.31}
$$

which substituted in (5.30) allows us to define $\text{Var}[K_{\mathcal{T}}]$ as:

$$
\text{Var}[K_{\mathcal{T}}] = \frac{p_{\mathcal{T}}(t)}{\mathcal{T}} \int_{\mathbb{R}} k_{\mathcal{T}}(\psi)^2 \, d\psi - p_{\mathcal{T}}(t)^2. \tag{5.32}
$$

These transformations allow us to express (5.29) as:

$$
\begin{aligned}
\text{Var}[\epsilon] &= \frac{1}{M}(\text{Var}[\hat{I}] + \text{E}[\hat{I}]^2) \\
&\quad (\frac{p_{\mathcal{T}}(t)}{\mathcal{T}} \int_{\mathbb{R}} k_{\mathcal{T}}(\psi)^2 \, \mathrm{d}\psi - p_{\mathcal{T}}(t)^2 + p_{\mathcal{T}}(t)^2) \\
&\quad - \frac{1}{M} p_{\mathcal{T}}(t)^2 \text{E}[\hat{I}]^2 \\
&= \frac{1}{M}(\text{Var}[\hat{I}] + \text{E}[\hat{I}]^2)(\frac{p_{\mathcal{T}}(t)}{\mathcal{T}} \mathcal{C}_{\mathcal{T}})
\end{aligned}
\tag{5.33}
$$

where $\mathcal{C}_{\mathcal{T}}$ is a kernel-dependent constant. The last term can be neglected by assuming that the kernels cover small areas in their respective domains, which effectively means that $\mathcal{C}_{\mathcal{T}} \gg p_{\mathcal{T}}(t)$. Equation (5.33) shows that in transient density estimation, the variance $\text{Var}[\epsilon]$ is inversely proportional to $\mathcal{T}$.

BIAS   Bias at each iteration j is defined as the expected value of the error $\text{E}[\epsilon_j]$:

$$
\begin{aligned}
\text{E}[\epsilon_j] &= \text{E}[\frac{1}{M}\sum_{i=1}^{M} K_{\mathcal{T}}(\|t - T\|)\hat{I} - I] \\
&= \text{E}[K_{\mathcal{T}}(\|t - T\|)]\text{E}[\hat{I}] - I.
\end{aligned}
\tag{5.34}
$$

As with the variance, we need to assume that the samples time and contribution can be interpreted as independent identically distributed random samples from the random variables T (with probability density $p_{\mathcal{T}}(t)$) and $\hat{I}$ respectively.

The expected value of $K_{\mathcal{T}}(\|t - T\|)$ is described as:

$$
\begin{aligned}
\text{E}[K_{\mathcal{T}}] &= \int_{\Omega_{\mathcal{T}}} K_{\mathcal{T}}(\|t - T\|) p_{\mathcal{T}}(T) \, \mathrm{d}T \\
&= \int_{\mathbb{R}} \frac{1}{\mathcal{T}} k_{\mathcal{T}}(\|t - T\|/\mathcal{T}) p_{\mathcal{T}}(T) \, \mathrm{d}T.
\end{aligned}
\tag{5.35}
$$

The second form of the equation is obtained by transforming $K_{\mathcal{T}}$ into a unit canonical form of the kernel $k_{\mathcal{T}}$ such that $K_{\mathcal{T}}(\xi) = k_{\mathcal{T}}(\xi/\mathcal{T})1/\mathcal{T}$. Assuming a locally uniform distribution $p_{\mathcal{T}}(\xi)$ (similar to modeling $\text{Var}[\epsilon]$) is too restrictive to model the expected error accurately, since it leads to zero bias in the limit. Therefore, following previous work [121], we use a Taylor expansion of $p_{\mathcal{T}}(\xi)$:

$$
p_{\mathcal{T}}(\xi) = p_{\mathcal{T}}(t) + (\xi - t)\nabla p_{\mathcal{T}}(t) + O(\|\xi - t\|^2).
\tag{5.36}
$$

We plug this expression into (5.35), and apply the changes of variable $\psi = (T - t)/\mathcal{T}$ and $\mathrm{d}T = \mathcal{T} \, \mathrm{d}\psi$, to get:

$$
\begin{aligned}
\text{E}[K_{\mathcal{T}}] &= \frac{1}{\mathcal{T}} \int_{\mathbb{R}} k_{\mathcal{T}}(\psi)(p_{\mathcal{T}}(t) + \mathcal{T}\psi\nabla p_{\mathcal{T}}(t) + O(\|\mathcal{T}\psi\|^2))\mathcal{T} \, \mathrm{d}\psi \\
&= p_{\mathcal{T}}(t) \int_{\mathbb{R}} k_{\mathcal{T}}(\psi) \, \mathrm{d}\psi + \mathcal{T}\nabla p_{\mathcal{T}}(t) \int_{\mathbb{R}} k_{\mathcal{T}}(\psi)\psi \, \mathrm{d}\psi \\
&\quad + \mathcal{T}^2 \int_{\mathbb{R}} k_{\mathcal{T}}(\psi)O(\|\psi\|^2) \, \mathrm{d}\psi \\
&\approx p_{\mathcal{T}}(t) + \mathcal{T}^2 \int_{\mathbb{R}} k_{\mathcal{T}}(\psi)O(\|\psi\|^2) \, \mathrm{d}\psi \\
&= p_{\mathcal{T}}(t) + \mathcal{T}^2 \mathcal{C}_{\mathcal{T}}^{ii}.
\end{aligned}
\tag{5.37}
$$

This means that bias due to radiance estimation in the temporal domain is inversely proportional to $\mathcal{T}^2$ and a constant $\mathcal{C}_{\mathcal{T}}^{ii}$ dependent on the high-order derivatives of the probability densities. We apply the last approximation by observing that in most common scenarios $\int_{\mathbb{R}} k_{\mathcal{T}}(\psi)\psi\,d\psi \approx 0$. Given this approximation, and using $I = p_{\mathcal{T}}(t)\mathrm{E}[\hat{I}]$, we can formulate the expected value $\mathrm{E}[K_{\mathcal{T}}]$ as:

$$\begin{aligned} \mathrm{E}[\epsilon_j] &\approx (p_{\mathcal{T}}(t) + \mathcal{T}^2\mathcal{C}_{\mathcal{T}}^{ii})\mathrm{E}[\hat{I}] - p_{\mathcal{T}}(t)\mathrm{E}[\hat{I}] \\ &= \mathrm{E}[\hat{I}]\mathcal{T}^2\mathcal{C}_{\mathcal{T}}^{ii}. \end{aligned} \tag{5.38}$$

5.A.2 *Variance and expected error of the pixel estimate*

Here we derive the variance and expected error of the pixel estimate $\langle I_n \rangle$ after the n-th pass of the progressive algorithm, as modeled in Equation (5.17). The sampled variable is the time instant t_j where the estimation is being computed. As previously, we assume that they are independent identically distributed random samples.

VARIANCE Assuming the random variable ϵ_j, we model the variance of the estimator $\mathrm{Var}[\langle I_n \rangle]$ as [121]:

$$\mathrm{Var}[\langle I_n \rangle] = \frac{1}{n}\mathrm{Var}[I] + \frac{1}{n^2}\sum_{j=1}^{n}\mathrm{Var}[\epsilon_j]. \tag{5.39}$$

The first term is the usual Monte Carlo estimator, which vanishes with $O(n^{-1})$. The other three terms, however, are functions of the error ϵ_j. $\mathrm{Var}[\epsilon_n]$ is the variance of the average error, modeled as:

$$\mathrm{Var}[\epsilon_n] = \sum_{j=1}^{n}\frac{1}{n}\mathrm{Var}[\epsilon_j] = \frac{1}{n^2}\sum_{j=1}^{n}\mathrm{Var}[\epsilon_j]. \tag{5.40}$$

In oder to achieve consistency, we allow the variance of the expected error to increase at each iteration by a factor (5.10):

$$\frac{\mathrm{Var}[\epsilon_{j+1}]}{\mathrm{Var}[\epsilon_j]} = \left(\frac{j+1}{j+\alpha}\right). \tag{5.41}$$

We can model $\mathrm{Var}[\epsilon_n]$ as a function of the variance at the first iteration $\mathrm{Var}[\epsilon_1]$ as:

$$\mathrm{Var}[\epsilon_n] = \frac{\mathrm{Var}[\epsilon_1]}{n^2}\left(1 + \sum_{j=2}^{n}j\alpha\mathrm{B}(\alpha,j)\right), \tag{5.42}$$

where $\mathrm{B}(x,y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)}$ is the Beta function, and $\Gamma(n) = (n-1)!$ is the Gamma function. Using the approximation proposed by Kaplanyan and Dachsbacher [113], we get:

$$\mathrm{Var}[\epsilon_n] \approx \frac{\mathrm{Var}[\epsilon_1]}{(2-\alpha)n^\alpha} = O(n^{-\alpha}). \tag{5.43}$$

Finally, using this formulation of the variance of the average error $\mathrm{Var}[\epsilon_n]$ and asypmtotic simplifications, we can formulate $\mathrm{Var}[\langle I_n \rangle]$ (5.39) as:

$$\begin{aligned} \mathrm{Var}[\langle I_n \rangle] &\approx \frac{1}{n}\mathrm{Var}[I] + \mathrm{Var}[\epsilon_n] \\ &\approx \frac{1}{n}\mathrm{Var}[I] + \frac{\mathrm{Var}[\epsilon_1]}{(2-\alpha)n^\alpha} \\ &= O(n^{-1}) + O(n^{-\alpha}) = O(n^{-\alpha}). \end{aligned} \tag{5.44}$$

EXPECTED ERROR   The expected value of of the estimator $\mathrm{E}[\langle I_n \rangle]$ is modeled as:

$$
\begin{aligned}
\mathrm{E}[\langle I_n \rangle] &= \mathrm{E}[\frac{1}{n} \sum_{j=1}^{n} (I_j + \epsilon_j)] \\
&= \frac{1}{n} \sum_{j=1}^{n} \mathrm{E}[I_j] + \frac{1}{n} \sum_{j=1}^{n} \mathrm{E}[\epsilon_j] \\
&= I + \mathrm{E}[\epsilon_n]
\end{aligned}
\tag{5.45}
$$

where $\mathrm{E}[\epsilon_n]$ is the bias of the estimator after n steps:

$$
\mathrm{E}[\epsilon_n] = \frac{1}{n} \sum_{j=1}^{n} \mathrm{E}[\epsilon_j],
\tag{5.46}
$$

and $\mathrm{E}[\epsilon_j]$ is the expected value of the error at each pass, described in Equation (5.38).

Following Equation (5.10) we compute $\mathcal{T}_j$ as a function of its initial value $\mathcal{T}_1$ as:

$$
\begin{aligned}
\mathcal{T}_j &= \mathcal{T}_1 \prod_{k=1}^{j-1} \left(\frac{k+\alpha}{k+1} \right) \\
&= \mathcal{T}_1 \left(\frac{(\alpha+1)_{j-1}}{\Gamma(j-1)} \right) \\
&= \mathcal{T}_1 \left(\frac{\Gamma(\alpha+j)}{\Gamma(\alpha+1)\Gamma(j+1)} \right) \\
&= \mathcal{T}_1 \left(\frac{\Gamma(\alpha+j)}{j \cdot \alpha \cdot \Gamma(\alpha)\Gamma(j)} \right) \\
&= \mathcal{T}_1 (j \cdot \alpha \cdot \mathrm{B}(\alpha,j))^{-1}
\end{aligned}
\tag{5.47}
$$

where $\Gamma(n)$ is the Gamma function, $\mathrm{B}(x,y)$ is the Beta function, and $(x)_n$ is the Pochhammer symbol $(x)_n = x(x-1)(x-2)..(x-n+1)$.

Using (5.47) we can express $\mathrm{E}[\epsilon_j]$ (5.38) as a function of the initial kernel bandwidths:

$$
\mathrm{E}[\epsilon_j] = \mathcal{T}_1^2 (j \cdot \alpha \cdot \mathrm{B}(\alpha,j))^{-2} \mathcal{C}_\mathcal{T}^{ii}.
\tag{5.48}
$$

As noted by Knaus and Zwicker [121], we can use Stirling's formula to get the asymptotic approximation of $\mathcal{T}_j$, which allows us to express $\mathrm{E}[\epsilon_j]$ in asymptotic notation:

$$
\mathrm{E}[\epsilon_j] = \mathcal{C}_\mathcal{T}^{ii} \mathcal{T}_1^2 \Theta(j^{1-\alpha})^{-2}.
\tag{5.49}
$$

Finally, we use $\sum_{j=1}^{n} \Theta(j^x) = nO(n^x)$ to plug Equation (5.49) into Equation (5.46) to get the asymptotic behavior of $\mathrm{E}[\epsilon_n]$:

$$
\begin{aligned}
\mathrm{E}[\epsilon_n] &= \frac{1}{n} \mathcal{C}_\mathcal{T}^{ii} \mathcal{T}_1^2 n O(n^{1-\alpha})^{-2} \\
&= O(n^{1-\alpha})^{-2}.
\end{aligned}
\tag{5.50}
$$

Inserting this last equality into Equation (5.45) allows us to compute the asymptotic form of $\mathrm{E}[\langle I_n \rangle]$ as:

$$
\mathrm{E}[\langle I_n \rangle] = O(n^{1-\alpha})^{-2}.
\tag{5.51}
$$

5.A.3 *Minimizing the Asymptotic Mean Squared Error*

As shown above, the convergence rates of both the variance $\text{Var}[\langle I_n \rangle]$ and the expected error $\text{E}[\langle I_n \rangle]$ depend on the value assigned to the parameter α. Ideally, we want to obtain the parameter that allows reducing faster the total error. We measure the total error using the asymptotic mean squared error (AMSE), defined in Equation (5.18); using the obtained values for variance $\text{Var}[\langle I_n \rangle]$ and bias $\text{E}[\langle I_n \rangle]$, it becomes:

$$AMSE(\langle I_n \rangle) = O(n^{-\alpha}) + O(n^{1-\alpha})^{-4}. \tag{5.52}$$

By finding the derivate of Equation (5.52) and equating to zero we get the optimal parameter $\alpha = 4/5$, which leads to the optimal convergence rate of the AMSE of our transient density estimate:

$$AMSE(\langle I_n \rangle) = O(n^{-\frac{4}{5}}) + O(n^{1-\frac{4}{5}})^{-4} = O(n^{-\frac{4}{5}}). \tag{5.53}$$

5.B TRANSIENT PROGRESSIVE PHOTON MAPPING

Here we describe our transient formulation of progressive photon mapping (PPM) [68, 67]. We first give an overview of the algorithm, describing the spatio-temporal smoothing kernel used, the progressive approach followed to vanish error in the limit, and the behaviour of the algorithm (Section 5.B.1). Then, we analyze the variance and expected error introduced by the spatio-temporal density estimation for a given iteration j (Section 5.B.2), and use these results to derive the asymptotic convergence rate after n iterations (Section 5.B.3). Finally, we detail the derivation of the optimal parameters for higher asymptotic convergence with respect to the AMSE (Section 5.B.4).

5.B.1 *Algorithm*

To include the temporal domain in a photon mapping framework, we need to take into account the time delays described in Equation (5.14), while adding a temporal smoothing kernel $K_{\mathcal{T}}$ in the density estimation, similar to Cammarano and Jensen [16]. This results in an approximation of the radiance $\widehat{L_o}(\mathbf{x}, t)$ as:

$$\widehat{L_o}(\mathbf{x}, t) = \frac{1}{M} \sum_{i=1}^{M} K(\|\mathbf{x} - \mathbf{x}_i\|, \|t - t_i^-\|)\gamma_i, \tag{5.54}$$

where $K(\mathbf{x}, t) = K_R(\mathbf{x}) \cdot K_{\mathcal{T}}(t)$ and K_R is the spatial smoothing kernels with bandwidth R. We decompose K into K_R and $K_{\mathcal{T}}$ since they operate in separate domains, which might result into different kernel types; this is a typical approach in multivariate density estimation [196].

In our transient framework, Equation (5.7) now becomes:

$$\langle I_n \rangle = \frac{1}{n} \sum_{j=1}^{n} \Psi_j \widehat{L_o}(\mathbf{x}_j, t_j), \tag{5.55}$$

where $\Psi_j = \mathfrak{T}(\overline{\mathbf{x}}_{w,j})/p(\overline{\mathbf{x}}_{w,j})W_e(\mathbf{x}_{k,j} \to \mathbf{x}_{k-1,j})$ is the importance of the measurement, which is the contribution of the eye subpath $\overline{\mathbf{x}}_{w,j}$. Unfortunately, although variance is reduced, using this estimator introduces bias in the final solution due to the expected error ϵ_j in each iteration, which depends

on the size of the kernel estimation (as seen in Section 5.3). This means that the estimate of the pixel $\langle I_n \rangle$ is the sum of the actual value of the pixel I and the expected error $\mathrm{E}[\epsilon_n]$:

$$\mathrm{E}[\langle I_n \rangle] = I + \mathrm{E}[\epsilon_n] = \Psi_j L_o + \frac{1}{n} \sum_{j=1}^{n} \Psi_j \mathrm{E}[\epsilon_j]. \tag{5.56}$$

Section 5.B.2 derives the expression for the expected error $\mathrm{E}[\epsilon_j]$ for the case of transient rendering.

ELIMINATING BIAS.   To eliminate bias, a transient *progressive* approach can be used. As discussed in Section 5.3, progressive photon mapping uses a number of photon tracing passes, each providing an increasingly accurate solution. It can be shown (see Section 5.B.2 for the full derivation) that the variance of the density estimation in transient PPM is inversely proportional to $R^2 \mathcal{T}$:

$$\frac{\mathrm{Var}[\epsilon_{j+1}]}{\mathrm{Var}[\epsilon_j]} = \frac{R_j^2 \mathcal{T}_j}{R_{j+1}^2 \mathcal{T}_{j+1}} = \frac{j+1}{j+\alpha}. \tag{5.57}$$

Equation (5.57) shows that transient PPM involves progressively reducing two parameters, R and $\mathcal{T}$, referring to the spatial and temporal domains. We thus split the variance scaling factor $(j+1)/(j+\alpha)$ into two, one for each kernel bandwidth. This yields:

$$\frac{\mathcal{T}_{j+1}}{\mathcal{T}_j} = \left(\frac{j+\alpha}{j+1} \right)^{\beta_{\mathcal{T}}}, \qquad \frac{R_{j+1}^2}{R_j^2} = \left(\frac{j+\alpha}{j+1} \right)^{\beta_R} \tag{5.58}$$

where $\beta_{\mathcal{T}}$ and β_R are scalars in the range $[0,1]$ which control how much each term is to be scaled separately, with $\beta_{\mathcal{T}} + \beta_R = 1$.

ERROR ANALYSIS.   For our analysis on the error of the estimate, we use the AMSE metric (5.18) again. The progressive approach described above ensures that error tends to zero when $n \to \infty$. However, the two terms of the AMSE metric have different convergence rates, dependent on the parameters α, $\beta_{\mathcal{T}}$ and β_R. We model the variance of the estimate $\mathrm{Var}[\langle I_n \rangle]$ as [121]:

$$\mathrm{Var}[\langle I_n \rangle] = \frac{1}{n} \mathrm{Var}[\Psi L_o] + \frac{1}{n^2} \sum_{j=1}^{n} \mathrm{Var}[\Psi \epsilon_j]. \tag{5.59}$$

The first term is a Monte Carlo estimator, with known convergence rate $O(n^{-1})$, while the second is the variance of the average error $\mathrm{Var}[\epsilon_n]$, which converges with $O(n^{-\alpha})$. The detailed derivation of this result appears in Appendix 5.B.3:

$$\mathrm{Var}[\langle I_n \rangle] = O(n^{-1}) + O(n^{-\alpha}) = O(n^{-\alpha}). \tag{5.60}$$

Therefore, variance for transient PPM converges at the same rate as standard PPM, which for $\alpha = 1$ converges at the same rate as a Monte Carlo estimator. The parameters $\beta_{\mathcal{T}}$ and β_R have no influence on the convergence rate.

On the other hand, the expected average error in transient PPM $\mathrm{E}[\epsilon_n]$ is (see Section 5.B.3):

$$\mathrm{E}[\langle I_n \rangle] = \mathrm{E}[\frac{1}{n} \sum_{j=1}^{n} \Psi_j (L_{o,j} + \epsilon_j)] = I + \mathrm{E}[\frac{1}{n} \sum_{j=1}^{n} \Psi_j \epsilon_j]. \tag{5.61}$$

while its convergence rate is:

$$E[\langle I_n \rangle] = O(n^{1-\alpha})^{-2\beta_{\mathcal{T}}} + O(n^{1-\alpha})^{\beta_{\mathcal{T}}-1} + O(n^{1-\alpha})^{-(1+\beta_{\mathcal{T}})}. \qquad (5.62)$$

In this case, the choice of $\beta_{\mathcal{T}}$ (and in consequence β_R) has a crucial effect on the asymptotic bias reduction. By observing the behavior in the limit, we find that the optimal parameter is $\beta_{\mathcal{T}} = 1/3$, which means that $E[\langle I_n \rangle]$ converges with $O(n^{-\frac{2(1-\alpha)}{3}})$.

Given these convergence rates, and using the AMSE metric given by Equation (5.18), we can now obtain the optimal parameter α for progressive transient radiance estimation, $\alpha = 4/7$ (see Section 5.B.4). This means that the AMSE vanishes asymptotically with order $AMSE = O(n^{-\frac{4}{7}})$. This shows that including the temporal domain leads to a slower convergence rate than standard PPM ($O(n^{-2/3})$ [113]); since the additional temporal kernel effectively increases bias, reducing it requires increasing the variance more at each step. However, note that in order to use standard PPM we would need to combine it with e.g. the histogram; this leads to a much slower convergence of standard PPM in *transient rendering*.

5.B.2   *Variance and Expected Error of Transient Radiance Estimation*

Following again the recent probabilistic framework for the progressive photon mapping algorithm [121], we first analyze the variance and expected value of the error introduced by the radiance estimate at each iteration. This error ϵ is defined as the difference between the estimated radiance $\widehat{L_o}$ and the actual radiance L_o, at point $\mathbf{x}$ and time t. Using the spatial and temporal kernels K_R and $K_{\mathcal{T}}$, with bandwidths R and $\mathcal{T}$ respectively, we have:

$$\epsilon(\mathbf{x}, R, t, \mathcal{T}) = \frac{1}{M} \sum_{i=1}^{M} K_R(\|\mathbf{x} - \mathbf{x}_i\|) K_{\mathcal{T}}(\|t - t_i^-\|) \gamma_i - L_o(\mathbf{x}, t). \qquad (5.63)$$

VARIANCE   In order to compute the variance of the error $\text{Var}[\epsilon]$ we need to make a set of assumptions: First, we assume that the photons' probability density is constant within the kernel K_R in the spatial domain [121], and within $K_{\mathcal{T}}$ in the temporal domain. We denote these probabilities as $p_R(\mathbf{x})$ and $p_{\mathcal{T}}(t)$ respectively. We also assume that the photons' position $\mathbf{x}_i$, time t_i and energy contribution γ_i are independent samples of the random variables X, T and γ, respectively, where X and T have probability densities $p_R(\mathbf{x})$ and $p_{\mathcal{T}}(t)$. Finally, we assume that the random variables X, T and γ are mutually independent. We thus model $\text{Var}[\epsilon]$ as:

$$\begin{aligned} \text{Var}[\epsilon] &= \text{Var}[\frac{1}{M} \sum_{i=1}^{M} K_R(\|\mathbf{x} - X\|) K_{\mathcal{T}}(\|t - T\|) \gamma - L_o(\mathbf{x}, t)] \\ &= \frac{1}{M}(\text{Var}[K_R] + E[K_R]^2)(\text{Var}[K_{\mathcal{T}}] + E[K_{\mathcal{T}}]^2) \\ &\quad (\text{Var}[\gamma] + E[\gamma]^2) - \frac{1}{M} E[K_R]^2 E[K_{\mathcal{T}}]^2 E[\gamma]^2. \end{aligned} \qquad (5.64)$$

Here $E[K_R] = p_R(\mathbf{x})$ and $E[K_{\mathcal{T}}] = p_{\mathcal{T}}(t)$, the variance introduced by the temporal kernel $\text{Var}[K_{\mathcal{T}}]$ is modeled by Equation (5.30), and $\text{Var}[K_R]$ is derived analogously to Equation (5.30) (see Section 5.A.1) as:

$$\text{Var}[K_R] = \frac{p_R(\mathbf{x})}{R^2} \int_{\mathbb{R}^2} k_R(\psi)^2 \, d\psi - p_R(\mathbf{x})^2. \qquad (5.65)$$

These transformations allow us to express (5.64) as:

$$
\begin{aligned}
\mathrm{Var}[\epsilon] \;=\; & \frac{1}{M}(\mathrm{Var}[\gamma] + \mathrm{E}[\gamma]^2) \\
& (\frac{p_R(\mathbf{x})}{R^2}\int_{\mathbb{R}^2} k_R(\psi)^2\,\mathrm{d}\psi - p_R(\mathbf{x})^2 + p_R(\mathbf{x})^2) \\
& (\frac{p_{\mathcal{T}}(t)}{\mathcal{T}}\int_{\mathbb{R}} k_{\mathcal{T}}(\psi)^2\,\mathrm{d}\psi - p_{\mathcal{T}}(t)^2 + p_{\mathcal{T}}(t)^2) \\
& -\frac{1}{M}p_R(\mathbf{x})^2 p_{\mathcal{T}}(t)^2 \mathrm{E}[\gamma]^2 \\
\approx\; & \frac{1}{M}(\mathrm{Var}[\gamma] + \mathrm{E}[\gamma]^2)(\frac{p_R(\mathbf{x})}{R^2}\mathcal{C}_R)(\frac{p_{\mathcal{T}}(t)}{\mathcal{T}}\mathcal{C}_{\mathcal{T}}) \qquad (5.66)
\end{aligned}
$$

where $\mathcal{C}_R$ and $\mathcal{C}_{\mathcal{T}}$ are kernel-dependent constants. The last term can be neglected by assuming that the kernels cover small areas in their respective domains, which effectively means that $\mathcal{C}_R \gg p_R(\mathbf{x})$ and $\mathcal{C}_{\mathcal{T}} \gg p_{\mathcal{T}}(t)$. Equation (5.66) shows that for transient density estimation, the variance $\mathrm{Var}[\epsilon]$ is inversely proportional to $R^2\mathcal{T}$. We show in Section 5.B.3 how this fact affects the shrinking formulation for progressive estimation.

BIAS   Bias at each iteration j is defined as the expected value of the error $\mathrm{E}[\epsilon_j]$:

$$
\begin{aligned}
\mathrm{E}[\epsilon_j] \;=\; & \mathrm{E}[\frac{1}{M}\sum_{i=1}^{M} K_R(\|\mathbf{x}-X\|)K_{\mathcal{T}}(\|t-T\|)\gamma - L_{\mathrm{o}}(\mathbf{x},t)] \\
\;=\; & \mathrm{E}[K_R(\|\mathbf{x}-X\|)]\mathrm{E}[K_{\mathcal{T}}(\|t-T\|)]\mathrm{E}[\gamma] - L_{\mathrm{o}}(\mathbf{x},t). \qquad (5.67)
\end{aligned}
$$

As with the variance, we need to assume that the photons' position, time and energy contribution can be interpreted as independent identically distributed random samples from the random variables X (with probability density $p_R(\mathbf{x})$), T (with probability density $p_R(\mathbf{x})$) and γ respectively.

The expected value of $K_{\mathcal{T}}(\|t-T\|)$ is described in Section 5.A.1, and modeled using Equation (5.37), while the expected value of the spatial kernel $\mathrm{E}[K_R]$ is derived in [121] as:

$$
\mathrm{E}[K_R] \approx p_R(\mathbf{x}) + R^2\int_{\mathbb{R}^2} k_R(\psi)O(\|\psi\|^2)\,\mathrm{d}\psi = p_R(\mathbf{x}) + R^2\mathcal{C}_R^{ii}. \qquad (5.68)
$$

Using Equations (5.37) and (5.68), and $L_{\mathrm{o}}(\mathbf{x},t) = p_R(\mathbf{x})p_{\mathcal{T}}(t)\mathrm{E}[\gamma]$ we get the expected value of the error $\mathrm{E}[\epsilon_j]$ for iteration j:

$$
\begin{aligned}
\mathrm{E}[\epsilon_j] \;\approx\; & (p_R(\mathbf{x}) + R^2\mathcal{C}_R^{ii})(p_{\mathcal{T}}(t) + \mathcal{T}^2\mathcal{C}_{\mathcal{T}}^{ii})\mathrm{E}[\gamma] \\
& -p_R(\mathbf{x})p_{\mathcal{T}}(t)\mathrm{E}[\gamma] \qquad\qquad\qquad\qquad (5.69) \\
\;=\; & \mathrm{E}[\gamma](p_R(\mathbf{x})\mathcal{T}^2\mathcal{C}_{\mathcal{T}}^{ii} + p_{\mathcal{T}}(t)R^2\mathcal{C}_R^{ii} + \mathcal{T}^2\mathcal{C}_{\mathcal{T}}^{ii}R^2\mathcal{C}_R^{ii}).
\end{aligned}
$$

5.B.3 *Variance and expected error of the pixel estimate*

Here we derive the variance and expected error of the pixel estimate $\langle I_n \rangle$ after n steps of the progressive algorithm, as modeled in Equation (5.56). The samples are the hit position $\mathbf{x}_j$ and time instant t_j where the radiance estimation is being computed. As previously, we assume that they are independent identically distributed random samples.

VARIANCE   Assuming that the random variables Ψ and ϵ_j are independent, we model the variance of the estimator $\text{Var}[\langle I_n \rangle]$ as [121]:

$$
\begin{aligned}
\text{Var}[\langle I_n \rangle] &= \frac{1}{n}\text{Var}[\Psi L_o] + \frac{1}{n^2}\sum_{j=1}^{n}\text{Var}[\Psi \epsilon_j] \\
&= \frac{1}{n}\text{Var}[\Psi L_o] + \text{Var}[\Psi]\frac{1}{n^2}\sum_{j=1}^{n}\text{Var}[\epsilon_j] \\
&\quad + \text{E}[\Psi]^2\frac{1}{n^2}\sum_{j=1}^{n}\text{Var}[\epsilon_j] + \text{Var}[\Psi]\frac{1}{n^2}\sum_{j=1}^{n}\text{E}[\epsilon_j]^2.
\end{aligned}
\tag{5.70}
$$

The first term is the usual Monte Carlo estimator, which vanishes with $O(n^{-1})$. The other three terms, however, are functions of the error ϵ_j. $\text{Var}[\epsilon_n]$ is the variance of the average error, modeled as:

$$
\text{Var}[\epsilon_n] = \sum_{j=1}^{n}\frac{1}{n}\text{Var}[\epsilon_j] = \frac{1}{n^2}\sum_{j=1}^{n}\text{Var}[\epsilon_j].
\tag{5.71}
$$

As described before, we allow the variance of the expected error to increase at each iteration by a factor:

$$
\frac{\text{Var}[\epsilon_{j+1}]}{\text{Var}[\epsilon_j]} = \left(\frac{j+1}{j+\alpha}\right)^{\beta_T}\cdot\left(\frac{j+1}{j+\alpha}\right)^{\beta_R} = \frac{j+1}{j+\alpha}.
\tag{5.72}
$$

Following [121], we can model $\text{Var}[\epsilon_n]$ as a function of the variance at the first iteration $\text{Var}[\epsilon_1]$ as:

$$
\text{Var}[\epsilon_n] = \frac{\text{Var}[\epsilon_1]}{n^2}\left(1 + \sum_{j=2}^{n} j\alpha \text{B}(\alpha, j)\right),
\tag{5.73}
$$

where $\text{B}(x, y)$ is the Beta function. This value can be approximated as [113]:

$$
\text{Var}[\epsilon_n] \approx \frac{\text{Var}[\epsilon_1]}{(2-\alpha)n^\alpha} = O(n^{-\alpha}).
\tag{5.74}
$$

Finally, using this formulation of the variance of the average error $\text{Var}[\epsilon_n]$ and asypmtotic simplifications, we can formulate $\text{Var}[\langle I_n \rangle]$ (5.71) as:

$$
\begin{aligned}
\text{Var}[\langle I_n \rangle] &\approx \frac{1}{n}\text{Var}[\Psi L_o] + \text{E}[\Psi]^2\text{Var}[\epsilon_n] \\
&\approx \frac{1}{n}\text{Var}[\Psi L_o] + \frac{\text{Var}[\epsilon_1]}{(2-\alpha)n^\alpha} \\
&= O(n^{-1}) + O(n^{-\alpha}) = O(n^{-\alpha}).
\end{aligned}
\tag{5.75}
$$

EXPECTED ERROR   The expected value of of the estimator $\text{E}[\langle I_n \rangle]$ is modeled as:

$$
\begin{aligned}
\text{E}[\langle I_n \rangle] &= \text{E}[\frac{1}{n}\sum_{j=1}^{n}\Psi_j(L_{o,j} + \epsilon_j)] \\
&= \frac{1}{n}\sum_{j=1}^{n}\text{E}[\Psi_j L_{o,j}] + \frac{1}{n}\sum_{j=1}^{n}\text{E}[\Psi_j]\text{E}[\epsilon_j] \\
&= I_n + \text{E}[\Psi]\text{E}[\epsilon_n]
\end{aligned}
\tag{5.76}
$$

where $\text{E}[\epsilon_n]$ is the bias of the estimator after n iterations (5.46), and $\text{E}[\epsilon_j]$ is the expected value of the error at each pass (5.70).

Following Equation (5.58) and (5.47) we compute $\mathcal{T}_j$ as a function of its initial value $\mathcal{T}_1$ as:

$$\mathcal{T}_j = \mathcal{T}_1(j\alpha\mathrm{B}(\alpha,j))^{-\beta_{\mathcal{T}}} \tag{5.77}$$

where $\mathrm{B}(x,y)$ is the Beta function. Analogously, we compute R_j^2 as a function of its initial value R_1^2:

$$R_j^2 = R_1^2(j\alpha\mathrm{B}(\alpha,j))^{-\beta_R}. \tag{5.78}$$

Using (5.77) and (5.78) we can express $\mathrm{E}[\epsilon_j]$ as a function of the initial kernel bandwidths:

$$
\begin{aligned}
\mathrm{E}[\epsilon_j] &= \mathrm{E}[\gamma]p_R(\mathbf{x})\mathcal{T}_1^2(j\alpha\mathrm{B}(\alpha,j))^{-2\beta_{\mathcal{T}}}\mathcal{C}_{\mathcal{T}}^{ii} \\
&+ \mathrm{E}[\gamma]p_{\mathcal{T}}(t)R_1^2(j\alpha\mathrm{B}(\alpha,j))^{-\beta_R}\mathcal{C}_R^{ii} \\
&+ \mathrm{E}[\gamma]\mathcal{T}_1^2 R_1^2(j\alpha\mathrm{B}(\alpha,j))^{-(2\beta_{\mathcal{T}}+\beta_R)}\mathcal{C}_{\mathcal{T}}^{ii}\mathcal{C}_R^{ii},
\end{aligned}
\tag{5.79}
$$

which using Stirling's formula to approximate $\mathcal{T}j$ and Rj allows us to express $\mathrm{E}[\epsilon_j]$ as:

$$
\begin{aligned}
\mathrm{E}[\epsilon_j] &= \mathrm{E}[\gamma]p_R(\mathbf{x})\mathcal{C}_{\mathcal{T}}^{ii}\mathcal{T}_1^2\Theta(j^{1-\alpha})^{-2\beta_{\mathcal{T}}} \\
&+ \mathrm{E}[\gamma]p_{\mathcal{T}}(t)\mathcal{C}_R^{ii}R_1^2\Theta(j^{1-\alpha})^{-\beta_R} \\
&+ \mathrm{E}[\gamma]\mathcal{C}_{\mathcal{T}}^{ii}\mathcal{C}_R^{ii}\mathcal{T}_1^2 R_1^2\Theta(j^{1-\alpha})^{-(2\beta_{\mathcal{T}}+\beta_R)}.
\end{aligned}
\tag{5.80}
$$

Finally, we use $\sum_{j=1}^n \Theta(j^x) = nO(n^x)$ to plug Equation (5.80) into Equation (5.46) to get the asymptotic behavior of $\mathrm{E}[\epsilon_n]$ in transient progressive photon mapping:

$$
\begin{aligned}
\mathrm{E}[\epsilon_n] &= \frac{\mathrm{E}[\gamma]}{n}p_R(\mathbf{x})\mathcal{C}_{\mathcal{T}}^{ii}\mathcal{T}_1^2 nO(n^{1-\alpha})^{-2\beta_{\mathcal{T}}} \\
&+ \frac{\mathrm{E}[\gamma]}{n}p_{\mathcal{T}}(t)\mathcal{C}_R^{ii}R_1^2 nO(n^{1-\alpha})^{-\beta_R} \\
&+ \frac{\mathrm{E}[\gamma]}{n}\mathcal{C}_{\mathcal{T}}^{ii}\mathcal{C}_R^{ii}\mathcal{T}_1^2 R_1^2 nO(n^{1-\alpha})^{-(2\beta_{\mathcal{T}}+\beta_R)} \\
&= O(n^{1-\alpha})^{-2\beta_{\mathcal{T}}} + O(n^{1-\alpha})^{-\beta_R} + O(n^{1-\alpha})^{-(2\beta_{\mathcal{T}}+\beta_R)},
\end{aligned}
\tag{5.81}
$$

which, by using the equality $\beta_R = 1 - \beta_{\mathcal{T}}$, becomes:

$$\mathrm{E}[\epsilon_n] = O(n^{1-\alpha})^{-2\beta_{\mathcal{T}}} + O(n^{1-\alpha})^{\beta_{\mathcal{T}}-1} + O(n^{1-\alpha})^{-(1+\beta_{\mathcal{T}})}. \tag{5.82}$$

Inserting this last equality into Equation (5.76) allows us to compute the asymptotic form of $\mathrm{E}[\langle I_n\rangle]$ as:

$$\mathrm{E}[\langle I_n\rangle] = O(n^{1-\alpha})^{-2\beta_{\mathcal{T}}} + O(n^{1-\alpha})^{\beta_{\mathcal{T}}-1} + O(n^{1-\alpha})^{-(1+\beta_{\mathcal{T}})}. \tag{5.83}$$

where the error vanishes in the limit since we impose that $\mathrm{E}[\epsilon_j] > \mathrm{E}[\epsilon_j]$. This consistency has a convergence rate dependent on both α (the common PPM parameter), and $\beta_{\mathcal{T}}$, which balances the shrinking rate of the spatial and temporal kernels.

5.B.4 *Minimizing the Mean Squared Error*

Here we obtain the parameters α and $\beta_{\mathcal{T}}$ that allows optimal consistency in terms of ASME (5.18) in transient progressive photon mapping. Using the

expressions obtained for the variance $\mathrm{Var}[\langle I_n \rangle]$ and expected error $\mathrm{Var}[\langle I_n \rangle]$, we model AMSE as:

$$\begin{aligned} AMSE(\langle I_n \rangle) \quad = \quad & O(n^{-\alpha}) + (O(n^{1-\alpha})^{-2\beta_\mathcal{T}} \\ & + O(n^{1-\alpha})^{\beta_\mathcal{T}-1} + O(n^{1-\alpha})^{-(1+\beta_\mathcal{T})})^2. \end{aligned} \quad (5.84)$$

which is a function of the parameters α and $\beta_\mathcal{T}$. Given that the variance is independent of $\beta_\mathcal{T}$, we first obtain the optimal value for this parameter that yields the highest convergence rate of the bias $\mathrm{E}[\epsilon_n]$. After differenciating Equation (5.82), applying asymptotic simplifications and equating to zero, we obtain the optimal value $\beta_\mathcal{T} = 1/3$. By plugging this value in Equation (5.82), we can express the AMSE as:

$$AMSE(\langle I_n \rangle) = O(n^{-\alpha}) + O(n^{-\frac{4}{3}(1-\alpha)}). \quad (5.85)$$

Differentiating Equation (5.85) and equating to zero we get the optimal parameter $\alpha = 4/7$, which results in the optimal convergence rate of the AMSE of our transient progressive photon mapping formulation:

$$AMSE(\langle I_n \rangle) = O(n^{-\frac{4}{7}}) + O(n^{-\frac{4}{3}(1-\frac{4}{7})}) = O(n^{-\frac{4}{7}}). \quad (5.86)$$

5.C DERIVATIONS FOR TIME SAMPLING

Here we derive the different pdf shown in Section 5.6 and illustrate how each of those pdf lead to a uniform distribution of path samples along the temporal domain.

5.C.1 *Sampling scattering distance in eye/light subpaths*

This section describes the derivation for the pdf chosen in Section 5.6.1 of the paper. We aim to find a pdf $p(r)$ for a single segment of the light subpath, such as the distribution of subpath vertices along the temporal domain is uniform. We define $p\left(\cup_{i=1}^\infty t_i\right)$ based on the per-vertex temporal location probability distribution $p(t_i)$ for all subpath vertices:

$$p\left(\cup_{i=1}^\infty t_i\right) = \sum_{i=1}^\infty p(t_i), \quad (5.87)$$

where $p(t_i)$ is defined based as the addition on $t_i = t(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i-1}) + t_{i-1}$ (propagation time plus temporal location of the previous sample). The probability distribution of an addition is the convolution of the probability distribution of the addends, so therefore

$$p(t_i) = \int_0^{t_i} p\big(t(\mathbf{x}_{i-1} \leftrightarrow \mathbf{x}_i)\big) p(t_{i-1}) dt_{i-1}, \quad (5.88)$$

$$p(t_1) = p\big(t(\mathbf{x}_0 \leftrightarrow \mathbf{x}_1)\big). \quad (5.89)$$

The probability distribution of the propagation time $p\left(t(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i-1})\right)$ is related to the scattering distance pdf $p(r)$ by the change of variable $r = \frac{c}{\eta} t(\mathbf{x}_{i-1} \leftrightarrow \mathbf{x}_i)$.

This temporal distribution $p\left(\cup_{i=1}^\infty t_i\right)$ should be uniform in time. We are obviously not letting the system to cast paths of infinite number of interactions. As stated in the paper, we reject samples when they get out of the sensor temporal window.

For this derivation we will move our calculations to Laplace space. We note the Laplace transform of a function f as $\mathcal{L}\{f\}$, and we are interested in the following properties of the Laplace transform:

$$\mathcal{L}\{af(t)\} = a\mathcal{L}\{f\}(s) \tag{5.90}$$

$$\mathcal{L}\{f(t) + g(t)\} = \mathcal{L}\{f\}(s) + \mathcal{L}\{g\}(s) \tag{5.91}$$

$$\mathcal{L}\{f(at)\} = \frac{1}{|a|}\mathcal{L}\{f\}(\frac{s}{a}) \tag{5.92}$$

$$\mathcal{L}\{\int_0^t f(\tau)g(t-\tau)d\tau\} = \mathcal{L}\{f\}(s) \cdot \mathcal{L}\{g\}(s) \tag{5.93}$$

$$\mathcal{L}\{u(t)\} = \frac{1}{s} \tag{5.94}$$

$$\mathcal{L}\{e^{-\alpha t} \cdot u(t)\} = \frac{1}{s+\alpha} \tag{5.95}$$

where $u(t)$ is the unit step function:

$$u(t) = \begin{cases} 0 & t < 0 \\ 1 & t \geq 0 \end{cases} \tag{5.96}$$

and s is the variable in the Laplace frequency domain.

Lets prove that the exponential distribution $p(r) = u(r)\lambda e^{-\lambda r}$ leads to a uniform distribution of samples in time $p\left(\cup_{i=1}^{\infty} t_i\right)$. We start by applying the Laplace transform to $p(r)$ based on the Laplace properties defined in (5.90) and (5.95):

$$\mathcal{L}\{\lambda e^{-\lambda r}\} = \frac{\lambda}{s^{(r)} + \lambda} \tag{5.97}$$

where $s^{(r)}$ is the Laplace frequency domain representation of r. For any propagation time $t(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i-1})$ we always apply the same scattering distance pdf $p(r)$ so we apply the change of variable:

$$\mathcal{L}\{p(t(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i-1}))\} = \frac{c}{\eta}\frac{\eta}{c}\mathcal{L}\{p\}(\frac{\eta}{c}s^{(\leftrightarrow)}) = \frac{\lambda}{\frac{\eta}{c}s^{(\leftrightarrow)} + \lambda} \tag{5.98}$$

where $t(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i-1})$ represents propagation time in this case as a change of variable from distance r, and $s^{(\leftrightarrow)}$ is the Laplace domain representation of $t(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i-1})$. This is applied for the first interaction,

$$\mathcal{L}\{p(t_1)\} = \frac{c}{\eta}\frac{\eta}{c}\mathcal{L}\{p\}(\frac{\eta}{c}s_i) = \frac{\lambda}{\frac{\eta}{c}s_1^{(t)} + \lambda}, \tag{5.99}$$

where $s_i^{(t)}$ is the Laplace domain representation of t_i. The subsequent interactions are obtained by applying the convolution property (5.93) to (5.88):

$$\mathcal{L}\{p(t_i)\} = \mathcal{L}\{p\}(s^{(\leftrightarrow)}) \cdot \mathcal{L}\{p\}(s_{i-1}^{(t)}). \tag{5.100}$$

We then apply recursively (5.100) so $\mathcal{L}\{p(t_i)\}$ actually becomes a simple power:

$$\mathcal{L}\{p(t_i)\} = \left(\mathcal{L}\{p\}(s^{(\leftrightarrow)})\right)^i = \left(\frac{\lambda}{\frac{\eta}{c}s^{(\leftrightarrow)} + \lambda}\right)^i \tag{5.101}$$

We then apply the Laplace transform to (5.87) using the property (5.91):

$$\mathcal{L}\{p\left(\cup_{i=1}^{\infty} t_i\right)\} = \sum_{i=1}^{\infty} \left(\frac{\lambda}{\frac{\eta}{c} s^{(\leftrightarrow)} + \lambda}\right)^i \tag{5.102}$$

which is in fact an infinite geometric series, which has the following analytical solution:

$$\mathcal{L}\{p\left(\cup_{i=1}^{\infty} t_i\right)\} = \frac{1}{1 - \frac{\lambda}{\frac{\eta}{c} s^{(\leftrightarrow)} + \lambda}} - 1 = \frac{c}{\eta} \frac{\lambda}{s^{(\leftrightarrow)}} \tag{5.103}$$

If we then apply the anti-transform by properties (5.90) and (5.94) we get:

$$p\left(\cup_{i=1}^{\infty} t_i\right) = \frac{c}{\eta} \lambda u(t) \tag{5.104}$$

which is uniform in time. We reject all paths with total duration t_k out of the rendered time window $(0, t_e)$ so $p\left(\cup_{i=1}^{\infty} t_i\right)$ never reaches infinite time nor infinite number of bounces. Notice that this derivation is analogous for the eye subpath.

5.C.2   *Sampling line-to-point shadow connections*

This section describes the derivation of the line-to-point shadow connection pdf described in Section 5.6.2 of the paper.

Any sampling strategy is usually based on a pdf that can be analytically integrated to get an invertible cdf. In the case of steady-state strategies, the pdf reduces variance by approximating the integrand (in this case path contribution, radiance) as close as possible. This approach, however, cannot be directly applied to transient rendering, where the uniformity of samples along the temporal dimension becomes quite relevant for the accuracy of the result.

The inverse cdfselects a sample according to a uniformly distributed random number $\xi \in [0, 1)$. The key idea of this sampling strategy is to preserve the uniformity of such random number when changing to the temporal domain. The time to be uniformly sampled is the propagation time $\{\mathbf{x}_i, \mathbf{x}_{i+1}, \mathbf{x}_{i+2}\}$. We name the distance from $\mathbf{x}_i$ to $\mathbf{x}_{i+1}$ as r_{i+1}, and r_{i+2} is the distance from $\mathbf{x}_{i+1}$ to $\mathbf{x}_{i+2}$. Therefore, the total propagation time for this connection is

$$t = \frac{\eta}{c}\left(r_{i+2} + r_{i+1}\right). \tag{5.105}$$

We start by expressing the total propagation time as a function of r_{i+2}:

$$t_{i+2}^-(r_{i+2}) = \frac{\eta}{c}\left(r_{i+2} + \underbrace{\sqrt{r_{i+2}^2 - 2r_{i+2}(\mathbf{l}\cdot\omega) + \mathbf{l}\cdot\mathbf{l}}}_{r_{i+1}}\right) + t_i + \Delta t_{i+1}, \tag{5.106}$$

where $\mathbf{l} = \mathbf{x}_i - \mathbf{x}_{i+2}$. Inverting this allows us to sample path locations for a specified temporal duration:

$$r_{i+2}(t_{i+2}^-) = \frac{(t_{i+2}^- - t_i - \Delta t_{i+1})^2 - \frac{\eta^2}{c^2}(\mathbf{l}\cdot\mathbf{l})}{2\frac{\eta}{c}(t_{i+2}^- - t_i - \Delta t_{i+1}) - 2\frac{\eta^2}{c^2}(\mathbf{l}\cdot\omega)}. \tag{5.107}$$

To obtain shadow connections uniformly distributed in a time range (t_a, t_b), we use Equation (5.107) as the inverse cdf

$$cdf^{-1}(\xi) = r_{i+2}(\xi(t_b - t_a) + t_a). \tag{5.108}$$

The normalized derivative of (5.106) is then the sampling pdf:

$$p(r_{i+2}) = \frac{\eta}{c(t_b - t_a)} \left(1 + \frac{r_{i+2} - (\mathbf{1} \cdot \omega)}{\sqrt{r_{i+2}^2 - 2r_{i+2}(\mathbf{1} \cdot \omega) + (\mathbf{1} \cdot \mathbf{1})}} \right), \tag{5.109}$$

Assuming that animation lies within the time range $(0, t_e)$, we set the connection time limits (t_a, t_b) to the temporal range left by the rest of the path for that connection:

$$t_a = t_i + t(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i+2}) \tag{5.110}$$

$$t_b = t_e - \Delta t_k - \left(\sum_{j=i+2}^{k-1} t(\mathbf{x}_j \leftrightarrow \mathbf{x}_{j+1}) + \Delta t_j \right). \tag{5.111}$$

5.C.3 *Angular sampling*

This section describes the derivation of the angular time sampling strategy discussed on Section 5.6.3 of the paper.

For the angular sampling pdf we follow a similar strategy than for a line-to-point shadow connection: we also aim to preserve the uniformity of the random number ξ when moving to the temporal domain. The propagation time between the vertices $\{\mathbf{x}_i, \mathbf{x}_{i+1}, \mathbf{x}_{i+1}\}$ is

$$t = \frac{\eta}{c}(r_k + r_{i+1}), \tag{5.112}$$

where r_{i+1} is the distance from $\mathbf{x}_i$ to $\mathbf{x}_{i+1}$ and r_k is the distance from $\mathbf{x}_{i+1}$ to $\mathbf{x}_k$. We express this propagation time as a function of θ

$$t(\theta) = \frac{\eta}{c} \left(r_{i+1} + \underbrace{\sqrt{(r_{i+1} \sin \theta)^2 + (|\mathbf{1}| - r_{i+1} \cos \theta)^2}}_{r_k} \right), \tag{5.113}$$

where $\mathbf{1} = \mathbf{x}_k - \mathbf{x}_i$. This can be simplified to:

$$t(\theta) = \frac{\eta}{c} \left(r_{i+1} + \sqrt{r_{i+1}^2 + |\mathbf{1}|^2 - 2r_{i+1}|\mathbf{1}| \cos \theta} \right). \tag{5.114}$$

Inverting (5.114) allows us to choose an angle for a specified temporal duration:

$$\theta(t) = \arccos \left(\frac{|\mathbf{1}|^2 + 2t\frac{c}{\eta} - \left(t\frac{c}{\eta}\right)^2}{2|\mathbf{1}|r_{i+1}} \right). \tag{5.115}$$

We use (5.115) as the base for the inverse cdf

$$cdf^{-1}(\xi) = \theta(\xi(t(\pi) - t(0)) + t(0)), \tag{5.116}$$

which can be expanded into

$$cdf^{-1}(\xi) = \arccos \left(\frac{|\mathbf{1}| - 2r_{i+1}^2 \xi^2 - 2\xi r_{i+1}(|\mathbf{1}| - 1)}{r_{i+1}|\mathbf{1}|} \right). \tag{5.117}$$

The normalized derivative of (5.114) is therefore our angular pdf:

$$p(\theta) = \frac{r_{i+1} \sin \theta}{2\sqrt{r_{i+1}^2 + |\mathbf{l}|^2 - 2r_{i+1}|\mathbf{l}| \cos \theta}}. \tag{5.118}$$

6

RELATIVISTIC EFFECTS FOR TIME-RESOLVED LIGHT TRANSPORT

This chapter describes a real-time framework that allows interactive visualization of relativistic effects for time-resolved light transport. We explore the effects occurring due to ultrafast camera motion on time-resolved light transport, including time dilation, light aberration, frequency shift and radiance accumulation, considering not only linear motion, but also acceleration and rotation of the camera. The main goal of this project is to visualize on a physically-plausible way the data captured and synthesized using the works described in Chapter 4 and Chapter 5 respectively.

This work was first published at the Spanish Conference in Computer Graphics (CEIG) 2013, where it was selected one of the two Best Papers. Then an extension of the paper was published at Computer Graphics Forum, and presented at the Eurographics Symposium on Rendering (EGSR) 2015.

A. Jarabo, B. Masia, A. Velten, C. Barsi, R. Raskar & D. Gutierrez
Rendering Relativistic Effects in Transient Imaging
Spanish Conference on Computer Graphics (CEIG) 2013, *Best Paper (1 in 2)*

A. Jarabo, B. Masia, A. Velten, C. Barsi, R. Raskar & D. Gutierrez
Relativistic Effects for Time-Resolved Light Transport
Computer Graphics Forum, *to appear*

6.1 INTRODUCTION

Analyzing and synthesizing light transport is a core research topic in computer graphics, computer vision and scientific imaging [63]. One of the most common simplifications, rarely challenged, is the assumption that the speed of light is infinite. While this is a valid assumption in most cases, it is certainly not true: light travels extremely fast, but with finite speed. In this work, we lift this assumption and explore the consequences of dealing with time-resolved data (finite speed of light), focusing on the effects predicted by special relativity when the camera moves at speeds comparable with the speed of light.

Transient imaging has recently emerged as a vibrant, exciting area of research. Being able to analyze light transport at picosecond scale has already helped gain a better understanding of the complexities of light propagation [225, 224], to approximate the shape of hidden objects [223] or reflectance properties [162, 160]. In this paper, we offer a novel contribution by visualizing relativistic effects of time-varying radiance. Beyond the pure scientific interest of advancing the field of relativistic visualization, our work has direct applications in education and even games (see for instance *OpenRelativity* from the MIT Game Lab [126]). Additionally, it can also contribute to recent works which have set the ground for a time-resolved framework for light transport simulation [98] described in Chapter 5.

Relativistic rendering is not new [83, 18, 236]. However, our particular time-resolved framework implies by definition that radiance is not constant in the temporal domain. In particular, we need to address how time affects light aberration, the Doppler effect and the searchlight effect typically
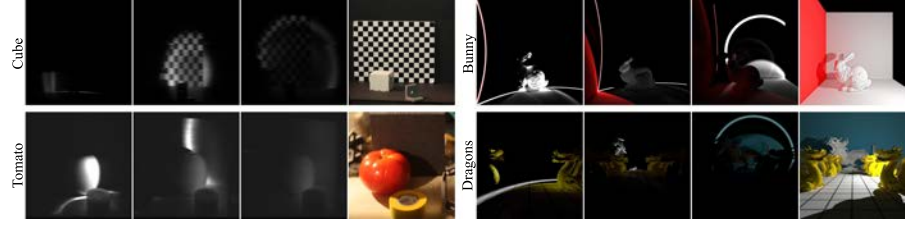
Figure 6.1: Scene setups and selected frames for the four scenes used throughout this chapter. The scenes on the left (*cube* and *tomato*) depict real data captured using femto-photography [224] (Chapter 4), while the scenes on the right (*bunny* and *dragons*) have been synthesized using time-resolved rendering [98] (Chapter 5).

handled in relativistic rendering, and include time dilation effects due to Lorentz contraction.

Additionally, we modify the traditional camera model used in previous literature and introduce for the first time—to our knowledge—a pinhole camera model in a relativistic rendering framework. This implies the consideration of two reference frames in the camera, co-moving with the pinhole and with the sensor respectively, and thus a focal length (distance pinhole-sensor) which can vary in certain reference frames under certain conditions; this, in turn, has an effect on the field of view of the camera, affecting the recorded imagery. We describe the required formulation for this camera model both in the presence of constant velocity and accelerated motion. Our model of relativistic acceleration is based on the clock hypothesis; we assume instantaneously co-moving frames of reference, thus being able to apply special relativity [30, 235]. Besides, we develop a model that accounts for relativistic rotation, allowing to freely rotate the camera.

We demonstrate our relativistic rendering framework with fly-throughs in a range of scenes. We rely on both captured data (using *femto-photography* [224], presented in Chapter 4, and which allows capturing light propagation at effective exposure times of less than 2 ps per frame), and synthesized data (using the transient rendering framework [101, 98] described in Chapter 5), which is shown in Figure 6.1.

In summary, we have developed a rendering and visualization tool for transient light transport, capable of simulating relativistic effects, lifting some of the assumptions and constraints of previous works. Our contributions can be summarized as follows:

- We describe a special relativity rendering framework which is able to handle time-resolved data

- We extend the traditional camera model used in the literature to a pinhole camera model, and describe the necessary transformations for rendering time-resolved scenes under this new model, both in the case of constant velocity and accelerated motion

- We propose an approximate solution for the case of relativistic *rotation*, including relativistic deformation of the sensor, so the camera can be moved freely in 3D space

6.2 RELATED WORK

TIME RESOLVED LIGHT TRANSPORT   A modified rendering equation can account for the finite speed of light and handle transient effects [7, 205]. However, in previous works no practical rendering framework is derived from the proposed transient rendering framework. A fully functional time-resolved rendering system was recently presented by Jarabo and colleagues [101, 98]. In their work, they introduce the transient path integral framework, and devise new sampling strategies and density estimation techniques for efficient rendering. Wu et al. [246] perform a rigorous analysis on transient light transport in frequency domain. They derive an analytic expression that models the information transfer between dimensions, and show that this derivation can be exploited to build a bare-sensor imaging system.

Time-resolved imaging is further analyzed by Wu et al. [245] to separate direct and global illumination components in macroscopic table-top scenes. The authors analyze the time profile for each pixel and decompose it into direct, subsurface scattering and interreflection components. Kirmani et al. [119] utilized global information in time-of-flight images to infer geometries of hidden objects, not directly visible by the camera, by using the encoded time-of-flight of diffuse indirect illumination. This work was further improved by Velten et al. [223]. Material BRDFs of distant patches were reconstructed [162] via light-bounce analysis from ultrafast image measurements.

Very recently, Velten et al. [225, 224] developed femto-photography, a novel technique that allows ultra-fast (in the order of picoseconds) capture of transient light transport, at an effective frequency of a trillion frames per second.

RELATIVISTIC RENDERING   Here we discuss the most relevant work on relativistic rendering. For a wider survey, we refer to [236], where the different proposed techniques for both general and special relativistic rendering are discussed, including their application as educational tools. A number of early works by Hsiung et al. [83] were the first to tackle rendering of relativistic effects in the field of computer graphics and visualization, applying the Lorentz transform to render scenes in which the observer moves at relativistic speeds relative to the scenes. A set of works followed, extending this work to three dimensional motion, considering the Doppler shift and time dilation, and improving rendering times [86, 84, 87, 85]. Later, Chang et al. [18] developed a more comprehensive framework accounting for both geometric and radiance transformations, including light aberration (Lorentz transform), the Doppler effect, and the searchlight effect. However, their formulation modeled the searchlight and Doppler effects incorrectly; these were corrected by Weiskopf et al. [238], who later further described the rendering pipeline in [233]. Weiskopf was also the first to formalize a complete rendering pipeline which could deal with special relativity and considered not only moving observers but also dynamic scenes, including acceleration, and moving light sources [235]. This pipeline, with an application to a virtual reality environment, is described in [234], while special relativity renderings of accelerated extended objects are considered in [236, 128][1]. Acceleration is further dealt with [239], correcting earlier work by Betts [9]. Related work simulates relativistic effects in synthetic scenes

---

1 English version available at http://www.spacetimetravel.org/tompkins/tompkins.html

modeled with image-based techniques, by applying the relativistic transformations directly on the light field [237]. Finally, visualization approaches and games have been created with a didactic goal, aiming at helping students in the understanding of relativity. Doat et al. [31] developed a billiard game, with a relativistic rendering engine and a non-Newtonian physics engine to deal with the collisions, to help students visualize the effects of special relativity. The game *A Slower Speed of Light*, also notable among these, uses the open-source toolkit *OpenRelativity* which works with the *Unity* engine and can also simulate special relativity effects [126]. In this paper we develop a relativistic rendering framework including a pinhole camera model, with a finite focal length; this has an effect on image formation, since, as we will show, this focal length can suffer contractions and dilations in the sensor frame of reference. Further, we show results of relativistic rendering of time-resolved data acquired from the real world.

6.3 RELATIVISTIC RENDERING

Time-resolved data allows us to explore light transport like never before, no longer being constrained by the assumption that light speed is infinite. However, current capture systems only allow to visualize light propagation from the capture point of view. Assuming that the geometry in the scene is known (which can be easily acquired with a digitizer arm or from time-of-flight data), we can synthesize new viewpoints and animations of the scene by taking an image-based rendering approach, using x-y textures from the x-y-t data cube (see [224] and Section 4.4) and projecting them onto the geometry. This allows us to visualize real-world events from new angles.

6.3.1 *Time-warping*

Visualizing light transport events at pico-second scale yields counter-intuitive results, as observed by Velten et al. [224] and described in Section 4.6. Due to the finite speed of light, events are not captured in the sensor as they occur, which leads to unexpected apparent distortions in the propagation of light. From this observation, it follows that different temporal frames of reference must be employed: one for the world O (when the events occur), and one for the camera sensor O' (when the events are actually captured).

As a consequence, sensor data acquired by the femto-photography technique appears *warped* in the temporal domain, and must be *time-unwarped* to take into account the finite speed of light. So for each frame in the synthesized animations, we access the original warped data and apply the following transformation [224]:

$$t'_{ij} = t_{ij} + \frac{z_{ij}}{c/\eta} \tag{6.1}$$

where t'_{ij} and t_{ij} are sensor (camera) and world times respectively, z_{ij} is the depth from each point (i, j) to the new camera position, c is the speed of light and η the index of refraction of the medium. Note how a naive approach based on simply sticking the textures from the first frame to the geometry through the animation would produce wrong results; the distance from each geometry point to the center of projection of the camera varies for each frame, and thus a *different* transformation must be applied each time to the original, *warped* x-y-t data (see Figure 6.2).

Figure 6.2: Time unwarping between camera time and world time for synthesized new views of a cube scene. Top row, left: Scene rendered from a novel view keeping the unwarped camera time from the first frame (the small inset shows the original viewpoint). Right: The same view, warping data according to the new camera position. Notice the large changes in light propagation, in particular the wavefronts on the floor not visible in the previous image. Bottom row: Isochrones visualization of the cube-scene for a given virtual camera (color encodes time); from left to right: original x-y-t volume in the time-frame of the capturing camera, unwarped x-y-t data in world time frame, and re-warped data for the new virtual camera.

6.3.2 *Camera Model*

We model our camera as a pinhole camera, with a focal distance d from the sensor to the pinhole aperture, and a field of view (FOV) $\alpha = 2\arctan(w/d)$, with w the half width of the sensor (Figure 6.3, left). We assume the camera has no lenses, the pinhole being what ensures correct focusing on the sensor. This model effectively splits the camera into two frames of reference: one frame O' which will be co-moving with the sensor, and one for the pinhole aperture O''. Note that considering the camera as a finite, instead of a point-like, object, has important effects on the recorded imagery due to relativistic contractions and dilations of the focal length, which will need to be taken into account. We detail how this is done in the following sections.

6.3.3 *Relativistic Effects*

In addition to the time-warping of data, which occurs in both static and dynamic cameras, macroscopic camera movement at pico-second time scales, like the one synthesized in Figure 6.2, would give rise to relativistic effects. This requires a relativistic framework to correctly represent and visualize light traveling through the 3D scene.

According to special relativity, light aberration, the Doppler effect, and the searchlight effect need to be taken into account when simulating motion at fast speeds. *Light aberration* accounts for the apparent geometry deformation caused by two space-time events measured in two reference frames moving at relativistic speeds with respect to each other. The *Doppler effect* produces a wavelength shift given by the Doppler factor. Last, the *searchlight effect* increases or decreases radiance, according to whether the observer is approaching or moving away from a scene.
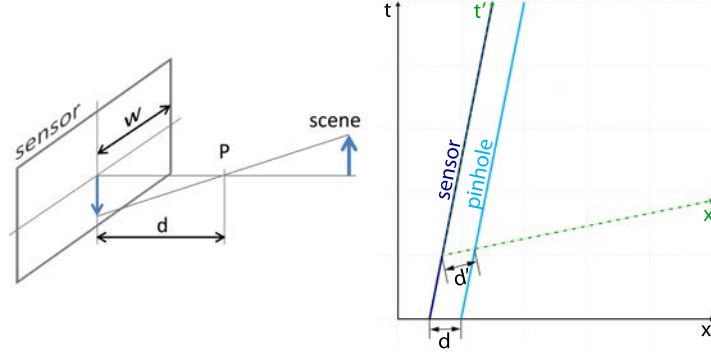
Figure 6.3: Left: Pinhole camera model, with P marking the position of the pinhole, w the half-width of the sensor, and d the focal length. Right: World lines for the sensor and the pinhole, moving with constant speed $\beta = 0.2$. The green dotted axes depict a local frame of reference of the sensor; these axes show how, in the frame of the sensor, the focal length d' is larger than when measured in the world frame, d. Note that the t axis actually corresponds to ct, but only t is stated in this and subsequent figures for clarity.

We build our relativistic visualization framework on the derivations by Weiskopf et al. [238]. We consider two inertial frames, O and O', where O' (the sensor) is moving with velocity $v = \beta c$ with respect to O, with $\beta \in [0.. \pm 1)$. L represents wavelength-dependent radiance[2] measured in O, defined by direction (θ, ϕ) and wavelength λ. θ and ϕ are defined with respect to the direction of motion, with $\theta \in [0..\pi]$ being the angle between the view and the motion vector, and $\phi \in [0..2\pi]$ the angular coordinate of the view vector around the motion vector on its tangent plane. The corresponding primed variables (θ', ϕ') and λ' define radiance L' measured in O'. To obtain the modified radiance L' given L and the speed of the sensor, we need to apply the following equation [238]:

$$L'(\theta', \phi', \lambda') = D^{-5} L \left(\arccos \frac{\cos \theta' + \beta}{1 + \beta \cos \theta'}, \phi', \frac{\lambda'}{D} \right), \tag{6.2}$$

where $D = \gamma(1 + \beta cos\theta')$ and $\gamma = 1/\sqrt{1 - \beta^2}$ are the Doppler and Lorentz factors, respectively. This equation accounts for all three factors: light aberration, the Doppler effect, and the searchlight effect. However, it cannot model explicitly the effect of special relativity on non-constant radiance. In the following paragraphs we explain each effect separately, and discuss the modifications needed to handle time-resolved radiance.

TIME DILATION   Breaking the assumption of constant radiance means that we cannot ignore the effect of time dilation [36]. Time dilation establishes that, given an observer moving with respect to a stationary (world) frame O, if Δt_0 is the elapsed time in the rest frame of the moving observer, the time elapsed for an observer in the stationary frame Δt, and the time elapsed for the moving observer are related by $\Delta t = \gamma \Delta t_0$. Since $\gamma \geq 1$, this effectively means that a clock carried by the moving observer will tick more slowly than a clock carried by the stationary observer. Consequently, if we take as the proper time the time in the frame O' in which the camera is at rest, the relationship between time intervals in the camera frame O' and the

---

2 For legibility we use the term "radiance" to describe "wavelength-dependent radiance" through the text.

world frame O is $\Delta t = \gamma \Delta t'$. Thus, we need to keep track of both world t and camera time t', since they differ depending on the speed of motion.

In addition to time dilation, Lorentz contraction shrinks the space in the direction of motion. This means that the time warping modeled with Equation (6.1), which accounts for the time it takes light to travel to the camera, has to be computed in the moving frame O'. Establishing the proper length for z in the world frame, the transformation to obtain z' is given by:

$$z' = z\sqrt{\sin^2 \theta + \frac{\cos^2 \theta}{\gamma^2}}. \tag{6.3}$$

CAMERA DEFORMATION   In addition to the effects on propagation delays, the Lorentz contraction also affects the focal length d of our pinhole camera (see Section 6.3.2). We define the proper length of the camera in its own rest frame O', thus being d' the proper focal length. Therefore, the focal length in the world frame O is $d = d'/\gamma$, as can be seen in the space-time diagram in Figure 6.3 (right). This effectively produces a reduction in the field of view of the camera α', such that:

$$\alpha' = 2\arctan\left(\frac{w}{\gamma d}\right) = 2\arctan\left(\frac{\tan(\alpha/2)}{\gamma}\right). \tag{6.4}$$

LIGHT ABERRATION   An easy example to understand light aberration is to visualize how we see rain drops when traveling on a speeding train. When the train is not moving, raindrops fall vertically; but as the train picks up speed, raindrop trajectories become increasingly diagonal as a function of the train's speed. This is because the speed of the train is comparable with the speed of raindrops. A similar phenomenon occurs with light if moving at relativistic speeds. However, as opposed to rain drops, relativistic light aberration cannot be modeled with classical physics aberration; the Lorentz transformation needs to be applied on top of it.

Light aberration is computed by transforming θ' and ϕ' with the following equations, which provide the geometric transformation between two space-time events measured in two reference frames which move at relativistic speeds with respect to each other:

$$\cos \theta' = \frac{\cos \theta - \beta}{1 - \beta \cos \theta} \tag{6.5}$$

$$\phi' = \phi. \tag{6.6}$$

The end result is that light rays are tilted towards the direction of motion as velocity increases, causing light rays reaching the sensor from behind the camera to become visible. Finally, as β approaches 1, and thus $v \approx c$, most incoming light rays are compressed towards the motion direction; this makes the scene collapse into a single point as the camera moves towards it (note that this produces the wrong impression that the camera is moving *away* from the scene). The first two rows in Figure 6.4 show the effects of light aberration with increasing velocity as the sensor moves at relativistic speeds, towards and away from the scene respectively.

DOPPLER EFFECT   The Doppler effect is better known for sound, and it is not a phenomenon restricted to relativistic velocities. In our case, the Doppler effect alters the observed frequency of the captured events in the
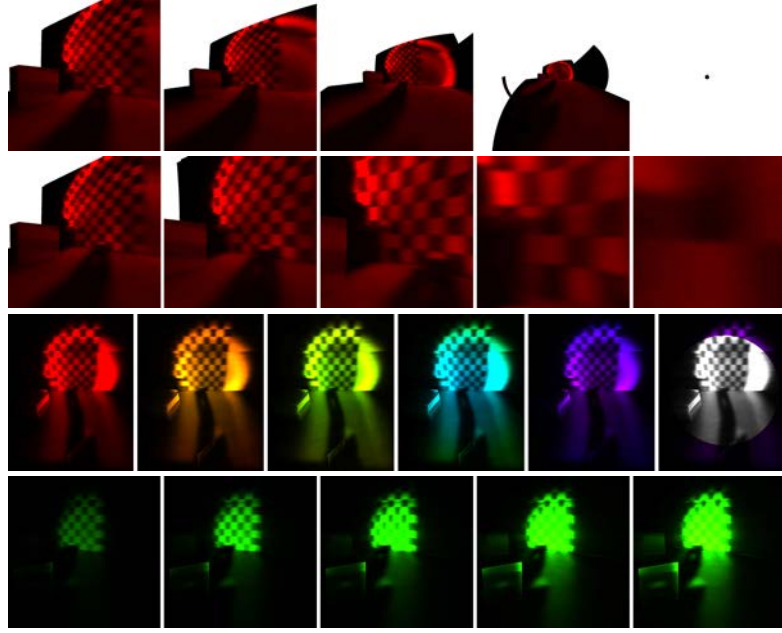
Figure 6.4: Relativistic effects shown separately for the *cube* scene. First rows: Distortion due to light aberration as the camera moves towards and away from the scene at different velocities, with $\beta = \{0, 0.3, 0.6, 0.9, 0.99\}$. We assume a laser wavelength of 670 nm for visualization purposes. Note how in both cases light aberration produces counter-intuitive results as the camera appears to be moving in the opposite direction. Third row: Doppler effect, showing the shift in color as a consequence of the frequency shift of light reaching the sensor, with $\beta = \{0, 0.15, 0.25, 0.35, 0.50, 0.55\}$. Fourth: Searchlight effect, resulting in an apparent increase in brightness as the speed of the approaching camera increases, with $\beta = \{0, 0.2, 0.3, 0.4, 0.5\}$ (simulated laser at 508 nm). All images have been tone-mapped to avoid saturation.
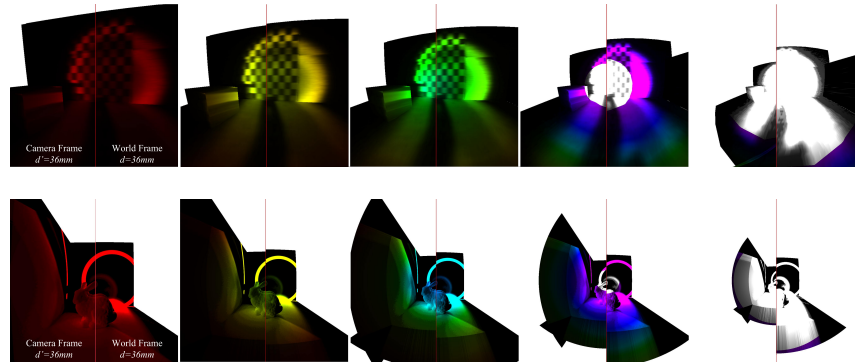
Figure 6.5: Relativistic phenomena for the *cube* (top, captured data) and *bunny* (bottom, simulated data) scenes including light aberration, Doppler effect and the searchlight effect, as the camera approaches the scene at increasing velocities $v = \beta c$, with β increasing from 0 to 0.77 (*cube*), and from 0.2 to 0.9 (*bunny*). In both scenes, we compare between fixing the focal distance d' in the sensor frame O' (left), and fixing it in the world frame O, and therefore being transformed in the moving frame (right). Note that we transform RGB radiance into luminance.

world when seen by a fast-moving camera, which produces a wavelength shift, as defined by the Doppler factor D:

$$\lambda' = D\lambda. \tag{6.7}$$

The overall result is a color shift as a function of the velocity of the sensor relative to the scene. Somewhat less known, the Doppler effect also creates a perceived speed-up (or down, depending on the direction of camera motion) of the captured events. This means that the *original frame rate* of the time-varying radiance f in world frame is Doppler shifted, making the perceived frame rate f' in camera frame become $f' = f/D$. Figure 6.4 (third row) shows an example of the Doppler effect.

SEARCHLIGHT EFFECT   Due to the searchlight effect, photons from several instants are captured at the same time differential, in part as a cause of the Doppler shift on the camera's perceived frame rate. This results in increased (if the observer is approaching the scene) or decreased (if the observer is moving away) brightness (see Figure 6.4, bottom row):

$$L'(\theta', \phi', \lambda') = D^{-5}L(\theta, \phi, \lambda). \tag{6.8}$$

Intuitively, continuing with our previous rain analogy, it is similar to what occurs in a vehicle driving in the rain: the front windshield will accumulate more water than the rear windshield. Equation (6.8) has an implicit dependence on time; for our time-varying streak-data, we need to make it explicit, yielding:

$$L'(\theta', \phi', \lambda', t') = D^{-5}L(\theta, \phi, \lambda, t). \tag{6.9}$$

In our case, since in the end we need to obtain the radiance accumulated over the duration of each frame of the movie, we are interested in computing the incoming radiance impinging the camera during a certain camera time interval $\mathbb{T}' = 1/f'$, where f' is the frame rate in the camera frame. As a consequence of the Doppler shift in the frame rate mentioned in the previous paragraph ($f' = f/D$), this camera time per frame $\mathbb{T}'$ corresponds to a time interval $\mathbb{T} = \mathbb{T}'/D$ in the world frame O. To do this, we need to integrate the wavelength-dependent radiance of Equation (6.9) over that time interval, resulting in the following:

$$\int_{\mathbb{T}'} L'(\theta', \phi', \lambda', t')dt' = \int_{\frac{\mathbb{T}'}{D}} D^{-5}L(\theta, \phi, \lambda, t)dt'. \tag{6.10}$$

When incorporating the previously described effects into Equation (6.10), we obtain the final expression for incoming radiance in the frame of the camera for each camera time frame (which has a duration $\mathbb{T}'$):

$$\int_{\mathbb{T}'} L'(\theta', \phi', \lambda', t')dt' =$$
$$D^{-5} \int_{\frac{\mathbb{T}'}{D}} L\left(\arccos \frac{\cos\theta' + \beta}{1 + \beta\cos\theta'}, \phi', \frac{\lambda'}{D}, \frac{t'}{\gamma} - \frac{z'}{c/\eta}\right) dt'. \tag{6.11}$$

Note that the D^{-5} factor used is valid only for a velocity vector v parallel to the sensor normal (see [238] for a detailed derivation); for other motion configurations that factor should be re-derived.

Finally, Figure 6.5 shows the result of combining all these relativistic effects, both for the *cube* scene (data captured with femto-photography techniques) and the *bunny* scene (simulated data by rendering) respectively, both

fixing the camera focal length in the world frame (i.e. varying d' with speed, as shown in Figure 6.3, right), and fixing it in the sensor frame. The laser wavelength is set at 670 nm for visualization purposes. We refer the reader to the supplementary videos to see the full animations.
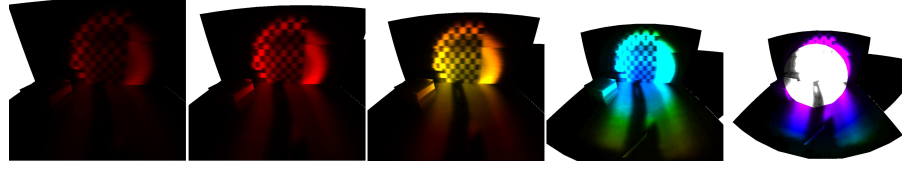
Figure 6.6: Relativistic phenomena for the *cube* scene due to an accelerating camera with focal length in world frame $d = 36$ mm. The camera starts at rest, and accelerates up to $v = 0.9c$ towards the scene, with acceleration $g = 0.01c$ s^{-1} in the camera frame O'. Note the slight differences in the field of view with respect to Figure 6.5.
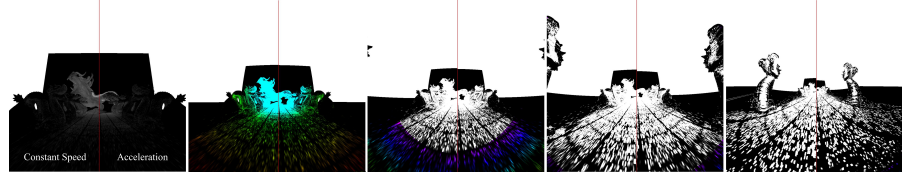
Figure 6.7: Comparison of taking into account the effect of relativistic acceleration on the camera (right), against the effects of using constant speed (left). The camera has a focal length $d = 36$ mm, and proper acceleration $g = 0.01c$ s^{-1}. Velocities shown here are $v = \beta c = [0.0..0.99c]$. Note that the simultaneity between the sensor and pinhole frames is broken in the scenario with acceleration, and therefore they have different speeds at proper time t'. As a consequence, the focal length d' in the moving frame O' is transformed differently in the presence of camera acceleration than when moving at constant speeds, an effect which becomes more visible at increasingly higher speeds.

6.3.4 *Relativistic Acceleration*

Until now, we have assumed that the camera accelerates instantaneously, and therefore no effect is produced by such acceleration. Similar to previous works [235, 30, 208], our framework for relativistic acceleration assumes that the clock hypothesis is valid. This hypothesis is implicit in the original formulation of special relativity, and states that the rate of a clock is only dependent on its *instantaneous* velocity. Therefore, an accelerating frame suffers the same relativistic effects as in its instantaneous co-moving inertial frame (i.e. a frame uniformly moving with the same instantaneous speed) [235], modeled using Equation (6.11). It is important to note that the clock hypothesis is not universally accepted [12, 44]; breaking the assumption of its validity makes the formulation of relativistic acceleration significantly more complex.

With this assumption, we model the acceleration a in the world frame O as [212]:

$$a' = \frac{dv'}{dt} = \frac{d}{dt}\gamma v = \gamma^3 a, \tag{6.12}$$

where a' is the constant proper acceleration in the moving frame; following the standard notation in physics literature, in the following we call it g. By
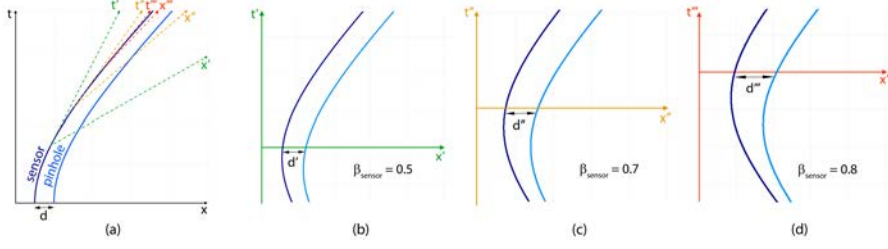
Figure 6.8: (a) World lines for the sensor and the pinhole in an accelerated scenario $(g = 0.002c \text{ s}^{-1})$, and local frames of reference of the sensor at different time instants and thus with different velocities (dotted axes). (b-d) Spacetime diagrams for the sensor and the pinhole in the three local frames of reference drawn in (a). Note how the focal length (distance between sensor and pinhole) measured in the sensor frame increases as the velocity of the sensor increases due to acceleration.

integration we can get the velocity in the world frame as a function of time t [208]:

$$v(t) = \frac{g(t - t_0)}{\sqrt{1 + (g(t - t_0)/c)^2}}, \tag{6.13}$$

where t_0 is the time at which acceleration starts ($v(t_0) = 0$). Integrating $v(t)$ allows us to obtain the position of the accelerating object in the world frame, $x(t)$, as:

$$x(t) = \frac{c^2}{g}\sqrt{1 + \left(\frac{g(t - t_0)}{c}\right)^2} - \frac{c^2}{g} + x_0, \tag{6.14}$$

where x_0 is the initial position of the object. This means that due to Lorentz contraction, the acceleration in the world frame is dependent on the velocity of the camera, which results in a hyperbolic motion with asymptotic speed c in the world frame O, as seen in Figure 6.8. Both v and x are defined as a function of time t in the world frame O, while we are capturing the scene from the sensor frame O', and therefore accounting for the proper time t'. These two times t and t' are related as:

$$t = \frac{c}{g}\sinh\left(\frac{gt'}{c}\right). \tag{6.15}$$

Note that Equations (6.13) and (6.14) are only valid for constant proper acceleration g; for time varying acceleration we would need to solve an initial value problem [234].

Acceleration has an important effect on our extended camera model, since an accelerating body changes its frame of reference at each instant; this, according to Equation (6.4), would make the focal distance d' increase as the object accelerates. However, when considering two different objects with the same acceleration and initial speed with respect to a world frame, we need to take into account that simultaneity is broken: two simultaneous instants in the world frame O might not be so in accelerating frame O'. The consequence of this is that the distance between the two objects (sensor and pinhole in our case) will be constant when measured in the world frame (Figure 6.8(a)), but will keep changing when measured in a frame local to either of the objects (as shown in Figure 6.8(b-d) for three different velocities of the sensor with respect to the world frame, corresponding to three different time instants along the world line of the sensor, depicted in Figure 6.8(a)).

Note that in our case, the distance between the two objects corresponds to the focal length (d in world frame O), which, when measured in the sensor frame O' (d'), is changing as the sensor (and pinhole) accelerates. This affects the field of view of the camera, and thus needs to be taken into account for the rendering framework.

We model this varying distance d' using the formulae derived by Styer [208]:

$$d' = \frac{c^2}{g} \left(1 - \sqrt{\frac{1 - (v/c)^2 + (gd/c^2)^2(v/c)^2}{1 - (v/c)^2}} \right) + \gamma d, \qquad (6.16)$$

where v is the speed of the sensor in the world frame (Equation (6.13)). Note that once both the sensor and pinhole stop accelerating, and therefore both are at constant speed βc, then $d' = \gamma d$, as explained in Section 6.3.3[3]. Equation (6.16) shows that the distance d' varies non-linearly with the distance d in the world frame; in fact, the larger the distance, the more notable are the effects due to an accelerating camera.

Given d', we can obtain the transformed field of view of the camera α' as $\alpha' = 2 \arctan(w/d')$ (for constant speed $g = 0$ it corresponds to Equation (6.4)). Note that in order to maintain a constant field of view α' during acceleration, different accelerations would be needed in the pinhole and the sensor, as derived by Franklin [43].

Figure 6.6 shows a sequence of an accelerating camera in the *cube* scene. We can see that, as opposed to the case in which a constant camera speed is used, the reduced field of view due to the Lorentz transform reduces the effect of light aberration observed with a camera at constant speed. This effect is more explicit in Figure 6.7, which compares the effect of constant speed and acceleration.
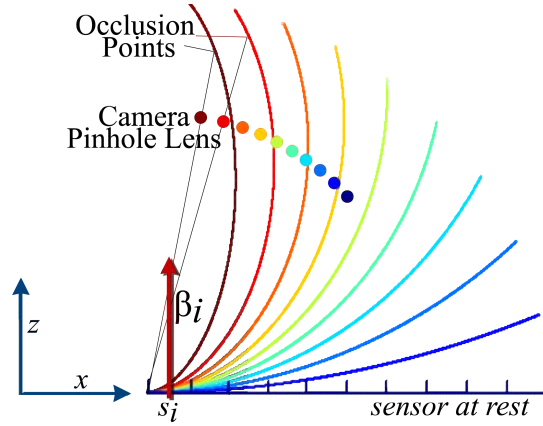
Figure 6.9: Distortion of the sensor (curves) and the camera pinhole lens (dots) due to relativistic rotation, as seen for the leftmost sensor differential area s_i. The velocity vector is perpendicular to such area (marked with a red arrow), with speeds ranging from $\beta_i = 0$ (dark blue, sensor at rest, no distortion), to $\beta_i = 0.9$ (dark red). Note how aberration in the sensor increases with higher speeds, even occluding the line at sight for very high values.

---

3 When this takes place, there will be a transition period in which, measured in the sensor frame, the sensor will stop accelerating before the pinhole; we have omitted this there for clarity but refer the interested reader to the thorough description in [208].
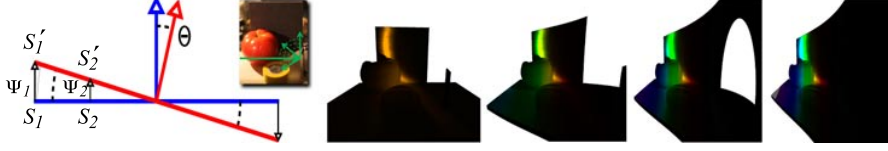
Figure 6.10: Assuming that the rotation angle θ can be neglected between frames, we model the rotation as a continuous linear velocity field on the sensor Ψ, so each differential area is assigned a different velocity ψ_s (left). This causes that depending on the position on the sensor, different relativistic transformations are applied on the scene. The rest of the frames show the effects of a clockwise rotation of the sensor, with $\beta = \{0, 0.4, 0.8, 0.99\}$ (measured at the edge of the sensor), assuming that no relativistic effect affects the camera. Figure 6.11 shows the effect when this last assumption is broken.

6.3.5 *Relativistic Rotation*

Providing free navigation of a scene depicting time-resolved light transport implies that the viewers should be allowed to rotate the camera. However, there is no universally accepted theory of relativistic rotation [189].

In the absence of a universal theory of relativistic rotation, we propose an approximation for the particular case of very small rotation angles per frame. This assumption, which is feasible in a scenario in which the camera moves at relativistic speeds, allows us to effectively treat each differential area of the sensor as undergoing linear translation only in the direction of the sensor's normal, with a different velocity for each sensor differential, according to the linear velocity field Ψ, and with a zero-crossing at the axis of rotation.

To simulate the rotation of the camera we therefore first divide the sensor S in different areas $s \in S$. Our approximation effectively turns each of them into a different translational frame, with linear velocity ψ_s. Then, for each s we render the scene applying the novel relativistic transformations introduced in this section, with a different β_s for each s (trivially obtained from an input β measured at the edge of the sensor). This causes the incoming radiance to be deformed differently depending on the position of the sensor where it is imaged. Figure 6.10 shows an example, where the sensor is rotating clockwise.

This transformation is enough if we are assuming that for each surface area s the camera lens moves with $\beta = \beta_i$, and that each sensor area s_i is independent of the rest. The center of the sensor thus has velocity $\beta = 0$, with each sensor area moving at different speeds. As a consequence, for an individual sensor area s_i, the sensor areas s_j suffer light aberration, each with different speed $\Delta\beta_{i\rightarrow j}$. This speed is the composition of β_i and β_j, and it is obtained using the relativistic law of composition of velocities:

$$\Delta\beta_{i\rightarrow j} = \frac{\beta_i + \beta_j}{1 + \beta_i\beta_j}. \tag{6.17}$$

This difference in speed makes both the camera lens and the sensor undergo light aberration, and therefore their projection is distorted following Equations (6.5) and (6.6). As a consequence, for each s_i with speed β_i, each ray from s_i to the camera lens is distorted, effectively changing the field of view at each sensor area. Additionally, since the rest of the sensor is aberrated with spatially-varying $\Delta\beta_{i\rightarrow j}$, it is even possible that the sensor plane
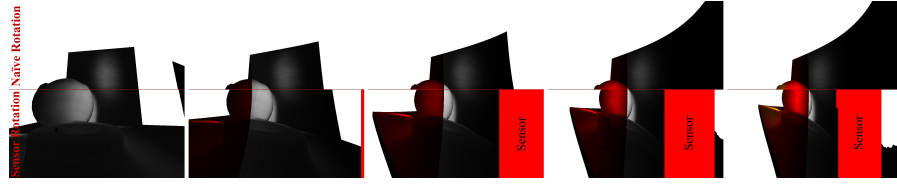
Figure 6.11: Relativistic effects of a clockwise rotation of the sensor, with $\beta = \{0, 0.25, 0.5, 0.75, 0.99\}$ (measured at the edge of the sensor), including relativistic aberrations in the camera. The lower half of the sequence shows the effect of rotation when taking into account the aberrations suffered by the sensor as seen from s_i (Figure 6.9), including the change in the FoV and the self-occlusion, while the upper half shows the rotation at the same speed ignoring its effect on the sensor. Note that at high-speeds, the edges of the sensor are occluded by the sensor itself due to sensor aberration, as shown in Figure 6.9 (marked here as a red rectangle). The image appearing to the right of this sensor occlusion comes from the scene, visible due to the high relativistic aberration caused by the rotation at high speeds.

itself occludes the incoming light, as shown in Figure 6.9. Figure 6.11 shows the same scene depicted in Figure 6.10, but including the aberration effects in the camera lens and sensor.

6.3.6 *Discussion*

In order to include the relativistic effects described above, there are several assumptions that need to be made in order to keep physical plausability. First, our work is restricted to special relativity, so we assume that no gravitational force is present in the scene; otherwise, general relativity would need to be considered. Additionally, we model relativistic acceleration based on the *clock hypothesis*.

We also need to assume that all components of the camera have infinite elasticity, to support the stress produced by relativistic acceleration and rotation, as well as a lensless pinhole camera. Including a lens would require adding lens transformations due to relativistic contractions (Sections 6.3.3 and 6.3.4), which would effectively change the plane of focus of the camera, and introduce additional time-varying delays. Our model does take into account the finite nature of the camera, including two frames of reference (pinhole and sensor), and a focal length, which can change under certain conditions. While in certain scenarios this effect may be negligible, it is important for time-varying radiance. Current transient data can reach a temporal resolution of picoseconds; at this timescale, a pinhole camera model makes a significant difference (Figure 6.7). To include rotation, we need to define a sensor with differential width, to avoid problems related with radius contraction as discussed in the Ehrenfest paradox [189]. Furthermore, for rotation we assume that the pinhole aperture is almost at the middle of the sensor, so it does not move when rotating. Finally, the formulae exposed here are constrained to camera motion parallel to the sensor normal; for other motion directions the equations modeling the Doppler and searchlight effect need to be modified (see [238] for the full mathematical derivation); in particular, the Doppler factor D should be re-derived accounting for the new configuration.

6.4 IMPLEMENTATION

Our implementation allows for real-time visualization of relativistic effects, both from real and simulated data. It is implemented in OpenGL as an stand-alone application, taking as input the reconstructed geometry of the scene, as well as the time-resolved data. The system is based on classic image-based rendering (IBR) techniques, where the shading of the surface is modeled by the images projected over the surface. In our case, we use x-y images from the x-y-t data cube to shade the geometry. The cube is stored as a 3D texture on the GPU in *world time* coordinates. This allows us to apply time-warping to adapt it to the new viewpoint in rendering time, by simply applying the transformation defined in Equation (6.1) (see Section 6.3.3).

Due to light aberration the geometry viewed from the camera is distorted. This distortion causes straight lines to become curved, so the geometry has to be re-tessellated. Image-space warping, which has been used in many scenarios (e.g. [237, 211, 149]) is not a viable alternative due to the large extent of the deformations. Our implementation performs the re-tessellation off-line on the CPU, but it is straightforward to tessellate it on the GPU on the fly. During render time, each vertex should be then transformed according to Equation (6.2). Another suitable alternative to the tessellation could be the use of 4D local ray tracing, such as the GPU-based system by Müller et al. [157].

The Doppler effect is introduced by modifying the wavelength of the outgoing illumination from the surfaces. To avoid the complexity of a full-fledged spectral renderer, we assume light with energy in only one wavelength of the spectrum. To display radiance we use a simple *wavelenght-to-RGB* conversion encoded as a 1D texture. Note that wavelengths out of the visible spectrum are displayed with gray-scale values in this paper.

Finally, when modeling the searchlight effect, we avoid the straightforward approach to access all frames in the x-y-t data cube, bounded by the duration $\mathbb{T}$, and integrate them. This would require several accesses to the 3D texture, which would hinder interactivity. Instead, we pre-integrate radiance in the temporal domain, and use anisotropic mipmapping to access the pre-integrated values, using $\mathbb{T}$ to select the mipmap level in the time dimension.

6.5 CONCLUSIONS AND FUTURE WORK

We have described a method to visualize light transport from a new perspective, no longer constrained by the assumption of infinite speed of light, and including relativistic effects with time-resolved light transport data (both real and simulated). Additionally, we have developed an interactive image-based rendering application for free navigation and exploration. We are able to account for both constant and accelerated motion, and have developed a method to approximate the effects of camera rotation, for which a definite solution does not exist in the physics literature. We have also introduced a pinhole camera model in a relativistic rendering framework, taking into consideration the variation of the focal length with camera motion at relativistic speeds, which in turn results in a varying field of view. We hope this will spur future research and help to better understand the complex behavior of time-resolved interactions between light and matter. All videos showing the effects presented here can be found at the project page[4].

---

4 http://giga.cps.unizar.es/~ajarabo/pubs/relativisticCGF15/

Of course there is plenty of future work ahead: Several assumptions discussed in Section 6.3.6 should be relaxed to generalize our model to other types of camera. Also, in order to provide a full physically-based solution mass (and therefore gravitational forces) must be included by considering general relativity. Our current implementation assumes Lambertian surfaces, so the viewing angle with respect to the normal has no influence in the result. This assumption can be relaxed by using more sophisticated IBR techniques, e.g. [10].

Part IV

CONCLUSION

CONCLUSIONS AND FUTURE WORK

7

In this thesis we have presented a variety of contributions on processing, synthesizing and visualizing multidimensional plenoptic data, departing from the two dimensional projection of the plenoptic function in traditional imaging. This higher dimensionality not only increases significantly the amount of information that can be obtained from scenes, but also changes drastically the way we handle visual data.

Through this thesis we have focused in the *angular* and *temporal* domains, each presenting different particularities but sharing the difficulty of handling and computing more visual data in a higher dimensional space than traditional imaging. In the former, we have focused on light fields, and reflectance fields described as BTFs, while in the temporal domain we have focused on time-resolved light transport, including capture and processing, synthesis and visualization. In the following we summarize the conclusions for each of the two parts.

ANGULAR DOMAIN   In this first part we have presented two main lines of work. In the first one (Chapter 2) we have presented a thorough analysis on how people interact with light fields when editing, focusing on point-based operations. For that, we have developed a fully-working interface for light field editing, and we have analyzed the two main interaction paradigms available in the literature (focus and parallax). We have performed a series of user studies, and collected information about both the preferences and workflows. Interestingly, we have found that the two main paradigms are complementary, and while focus is preferred for editing, parallax tends to be the choice for navigating through the light field, suggesting that a hybrid interface would be the most optimal for light field editing. Of course, this study cannot be considered final; instead it provides a solid foundation for the development of future interfaces for this multidimensional data. For example, more advanced edit workflows, such as lazy selection or sparse edit propagation are not analyzed in our study, where we focus only on point-wise operations: a first proof-of-concept for these types of interaction in light fields is briefly described in Section 2.7, but it definitely deserves more work. Another example operation not addressed in our work is copying and paste in light fields: this is particularly challenging, since in addition to the challenges existing on copy-paste in traditional two-dimensional images (e.g. maintaining illumination consistency), it introduces new such as correcting the angular-dependent component of the input light field into the target one. This requires new exploration on whether the angular inconsistencies are detected, how these affect the shape perception of the input light field, and also on how the angular domain of the input light field needs to be warped depending on its position to match the target one.

In the second work of this part we analyze the effect on the perception of the appearance of surfaces modeled using BTFs when filtering its reflectance field (Chapter 3). To perform this analysis, we conduct several large-scale perceptual studies, evaluating the effect of undersampling and blurring in both static and dynamic scenes, with different illumination and geometric setups. We then link our results with high- and low-level properties of sur-

faces, drawing a set of different applications from our results. Of course, as in all user studies, we do not claim that our work answers all questions regarding complex appearance filtering and perception; ours was the first work to systematically analyze the perceptual effect on material appearance of different BTF filtering strategies, and therefore several avenues of future work are still open. For example, more exploration into the relationships between appearance and the high-level visual properties might help understanding better the semantic effect on the perception of materials, even leading to better sampling strategies from this information, as shown in the sampling application described in Section 3.9. Another very interesting side finding is the correlation between the semantic high-level properties and low-level statistics on the material, which might have interesting applications on automatic material recognition based on image statistics; we believe that this last finding is a promising avenue for research to obtain a better understanding on why materials look as they do.

TEMPORAL DOMAIN   In this second part of the thesis we have focused on light transport at extremely high temporal resolution, so that the speed of light cannot longer be considered infinite. This has important implications on the technical challenges for capturing or simulating light transport, and introduces new problems on data processing, reconstruction or visualization. In particular, we have presented the first system capable of capturing light transport in macroscopic scenes at an effective temporal resolution of 2 ps (Chapter 4), in a project lead by colleagues at MIT Media Lab. This allows to visualize how light propagates through scene and interacts with matter. We have also shown that there are two different reference frames in transient light transport (world and camera frames), and proposed a method for *time unwarp* our data, to visualize the events as they happen, instead of as they are captured by the camera. Since the emergence of our work, several works have been proposed improving the temporal resolution up to tens of femtoseconds [52], or on the other side of the spectrum, reducing the hardware cost by trading-off temporal resolution [78, 111, 58]. Moreover, ultrafast imaging has been used not only for capturing stunning animations of light in motion. As happened with Edgerton's stroboscope in the 50's, several scientific and engineering applications have emerged from this new type of ultrafast data: classic problems in computer vision, such as depth recovery, can be easily solved by using ultrafast imaging [168, 161], while other problems very hard to solve, or even unsolvable, in traditional computer vision can now be tackled, such as non-line-of-sight depth reconstruction [223, 79] or motion estimation [171], illumination components decomposition [245, 88], BRDF capture [162, 160], imaging through turbid media [80] or single shot capture of scattering media [179].

This rapidly emerging field however lacked visualization and simulation tools helping on the development of new imaging and scene understanding techniques based on transient imaging. This motivated us to develop an interactive visualization system allowing to observe time-resolved light transport at different positions, taking in consideration the relativistic effects due to ultrafast motion of the camera (Chapter 6). Moreover, we addressed the lack of transient rendering tools by presenting a principled framework for time-resolved rendering (Chapter 5), which we use to demonstrate several non-trivial light transport effects, such as indirect illumination, caustics formation, fluorescence or birefringence. We reformulate the classic path integral formulation [218] into transient state, and based on that we propose a

method for efficiently reconstructing time-resolved radiance based on density estimation. We then propose a set of sampling techniques for light transport in participating media designed for sample uniformly the temporal domain, instead of focusing on radiance as in traditional steady-state sampling. This change of sampling domain, moving from sampling radiance to sampling the temporal domain results in a significantly better sample distribution along the simulation, and can inspire new sampling strategies for other applications where radiance is not the most important sampling domain. However, these techniques are only designed for participating media; in surfaces, the time-resolved radiance signal is fundamentally similar to a caustic, and therefore is extremely hard to sample the path's time-of-flight uniformly. Using a variant of manifold exploration [95] could significantly help, even in participating media, given that the propagation time of the full path could be sampled in a unified fashion. Apart from technical improvements the rendering, one of the main opened line of research is using time-resolved rendering in combination with capture methods, or as a tool helping scene understanding techniques. By using physically-accurate rendering we can develop a set of controlled priors for reconstruction-based transient imaging [78, 111], or even develop a dictionary of basis for sparse coding-based reconstruction, improving previous simple priors or basis (in general, only delta-functions and Gaussians are used [78, 88, 168, 180]). Inverse rendering [198] with time-resolved imaging is another powerful application for transient rendering, which given the extended dimensionality of the problem could allow even capturing visible and non-visible geometry and materials at once.

FUTURE WORK   Although in this thesis we have focused on only two domains of the plenoptic function, it does not mean that similar challenges as the addressed in our work for the angular and temporal domains are not present for the remaining plenoptic domains. For example, in a similar fashion as in light fields, is not well known nor established how to edit and manipulate hyperspectral images, which are gaining more importance in e.g. modern rendering systems, including production-oriented ones (e.g. Weta Digital's *Manuka* or Next Limit's *Maxwell^{TM}*). The dimensionality of the problem is even higher if we consider an extended plenoptic function, including wave properties such as polarization state or phase. Therefore, significant work following the same spirit of the work presented in this thesis is important to help on the development and wide-spread of this new type of multidimensional imaging.

Moreover, in this thesis we have explored the angular and temporal domain independently. This has allowed to simplify the problem, at the cost of integrating, and therefore losing, the other domain. Previous work has shown that there is a cross-dimensional information transfer between these two domains [247]. This suggests that handling jointly both domains can be beneficial when inferring information from the data, or at simulation time. This hypothesis is valid for the interaction between other domains: in a recent paper [97] we have seen that by simulating the angular and spatial domain jointly using adaptive rendering techniques we can reduce the rendering time of light fields around a 30% in simple scenes, and more as we increase the size and complexity in both domains. Other dimensions of the extended plenoptic function exhibit also close correlation: for example, the electromagnetic phase can be formulated as a function of the optical path (time) and the wavelength. Given these existing correlations between

different domains, an exciting avenue for future work is to explore the potential of including all, or a subset, of them for solving complex problems of scene understanding, and how to effectively manipulate, visualize or simulate them.

PERSONAL CONCLUSIONS   During the development of this thesis I have been able to work on a variety of projects, with different topics. This has allowed me to expand my knowledge, from pure rendering, which was my main field when started, to other fields such as perception, interaction or computational imaging. I think that is one of the most interesting aspects of the work I have done: instead of becoming extremely focused in one field I have also gained several insights from several related areas. Thus, although I still consider myself very knowledgeable in rendering, I think that open-mindedness is very valuable as a researcher, and it is probably a consequence of being in a rather multidisciplinary lab. I feel very lucky about that.

In addition to technical knowledge, during this years I have significantly improved other research skills, which are as valuable as pure mathematical or programming capabilities. Working with several collaborators from different institutions and backgrounds in most of my projects has led into significant evolution in terms of the way I communicate both problems and possible solutions, and also in work distribution and time management. This has helped to reach deadlines, and in general has made my life easier. Being able to supervise students has also had a deep impact on these skills. Also I have acquired some perspective, allowing me to see the projects as a whole, to put them in context or even to be able to kill them when they had to be killed. In this regard, the projects that died, or were put on hold *sine die*, have been as important as the projects that have made it in the thesis. Of course, my supervisor is to blame for most of these skills. Nevertheless, there is still a long way to go, and tons of stuff to learn and improve.

I must also mention the research stays and visits I have done during the thesis, that have given me the opportunity to learn from other groups, observing different lab dynamics and personalities, both in pure academia or in more industry-oriented research, and also acquire new knowledge and insights from new fields.

Finally, I would like to mention the most important thing of the thesis: I have been working in what I like, and having a lot of fun during the process. And of course it has been tough sometimes, but that only makes it more rewarding.

BIBLIOGRAPHY

[1] ABRAMSON, N. Light-in-flight recording by holography. *Optics Letters 3*, 4 (1978), 121–123.

[2] ADELSON, E. H., AND BERGEN, J. R. The Plenoptic function and the elements of early vision. *Computational Models of Visual Processing* (1991).

[3] AMENT, M., BERGMANN, C., AND WEISKOPF, D. Refractive radiative transfer equation. *ACM Trans. Graph. 33*, 2 (2014).

[4] AN, X., AND PELLACINI, F. Appprop: all-pairs appearance-space edit propagation. *ACM Trans. Graph. 27* (2008), 40:1–40:9.

[5] ANTANI, L., CHANDAK, A., TAYLOR, M., AND MANOCHA, D. Direct-to-indirect acoustic radiance transfer. *IEEE Trans. on Visualization and Computer Graphics 18*, 2 (2012).

[6] AO, H., ZHANG, Y., JARABO, A., MASIA, B., LIU, Y., GUTIERREZ, D., AND DAI, Q. Light field editing based on reparameterization. In *Pacific Rim Conference on Multimedia* (2015).

[7] ARVO, J. Transfer equations in global illumination. In *ACM SIGGRAPH '93 Courses* (1993).

[8] BERTRAM, M., DEINES, E., MOHRING, J., JEGOROVS, J., AND HAGEN, H. Phonon tracing for auralization and visualization of sound. In *IEEE Visualization '05* (2005).

[9] BETTS, C. Fast rendering of relativistic objects. *Journal of Visualization and Computer Animation 9*, 1 (1998), 17–31.

[10] BOIVIN, S., AND GAGALOWICZ, A. Image-based rendering of diffuse, specular and glossy surfaces from a single image. In *Proceedings of SIGGRAPH '01* (2001).

[11] BOUSSEAU, A., O'SHEA, J. P., DURAND, F., RAMAMOORTHI, R., AND AGRAWALA, M. Gloss perception in painterly and cartoon rendering. *ACM Trans. Graph. 32*, 2 (2013).

[12] BRILLOUIN, L. *Relativity reexamined*. Academic Press, 1970.

[13] BRUNETON, E., AND NEYRET, F. A survey of nonlinear prefiltering methods for efficient and accurate surface shading. *IEEE Trans. on Visualization and Computer Graphics 18* (2012).

[14] BURR, D. C., AND ROSS, J. Contrast sensitivity at high velocities. *Vision Research 22*, 4 (1982).

[15] BUSCK, J., AND HEISELBERG, H. Gated viewing and high-accuracy three-dimensional laser radar. *Applied Optics 43*, 24 (2004), 4705–4710.

[16] CAMMARANO, M., AND JENSEN, H. W. Time dependent photon mapping. In *Proceedings of EGWR '02* (2002).

[17] CAMPILLO, A., AND SHAPIRO, S. Picosecond streak camera fluorometry: a review. *IEEE Journal of Quantum Electronics 19*, 4 (1987), 585–603.

[18] CHANG, M.-C., LAI, F., AND CHEN, W.-C. Image shading taking into account relativistic effects. *ACM Trans. Graph. 15*, 4 (1996).

[19] CHARBON, E. Will avalanche photodiode arrays ever reach 1 megapixel? In *International Image Sensor Workshop* (2007), pp. 246–249.

[20] CHEN, B., OFEK, E., SHUM, H.-Y., AND LEVOY, M. Interactive deformation of light fields. In *Proceedings of I3D '05* (2005), pp. 139–146.

[21] COLAÇO, A., KIRMANI, A., HOWLAND, G. A., HOWELL, J. C., AND GOYAL, V. K. Compressive depth map acquisition using a single photon-counting detector: Parametric signal processing meets sparsity. In *Proceedings of CVPR '12* (2012), pp. 96–102.

[22] COLBERT, M., PATTANAIK, S., AND KŘIVÁNEK, J. Brdf-shop: Creating physically correct bidirectional reflectance distribution functions. *IEEE Comput. Graph. Appl. 26*, 1 (2006), 30–36.

[23] COLE, F., SANIK, K., DECARLO, D., FINKELSTEIN, A., FUNKHOUSER, T., RUSINKIEWICZ, S., AND SINGH, M. How well do line drawings depict shape? *ACM Trans. Graph. 28*, 3 (2009).

[24] CUNNINGHAM, D., AND WALLRAVEN, C. *Experimental Design: From User Studies to Psychophysics*. AK Peters, 2011.

[25] DANA, K. J., VAN GINNEKEN, B., NAYAR, S. K., AND KOENDERINK, J. J. Reflectance and texture of real-world surfaces. *ACM Trans. Graph. 18* (January 1999), 1–34.

[26] DANSEREAU, D. G. *Plenoptic signal processing for robust vision in field robotics*. PhD thesis, The University of Sydney, 2014.

[27] DAVIS, J., JÄHNE, B., KOLB, A., RASKAR, R., AND THEOBALT, C. Time-of-flight imaging: Algorithms, sensors and applications (Dagstuhl Seminar 12431). *Dagstuhl Reports 2*, 10 (2012).

[28] DEBEVEC, P. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *Proceedings of SIGGRAPH '98* (1998).

[29] D'EON, E., AND IRVING, G. A quantized-diffusion model for rendering translucent materials. *ACM Trans. Graph. 30*, 4 (2011).

[30] DEWAN, E. Stress effects due to Lorentz contraction. *American Journal of Physics 31* (1963).

[31] DOAT, T., PARIZOT, E., AND VÉZIEN, J.-M. Novative rendering and physics engines to apprehend special relativity. In *Proceedings of Eurographics Conference on Virtual Environments* (2011).

[32] DUGUAY, M. A., AND MATTICK, A. T. Pulsed-image generation and detection. *Applied Optics 10* (1971), 2162–2170.

[33] DURAND, F., HOLZSCHUCH, N., SOLER, C., CHAN, E., AND SILLION, F. X. A frequency analysis of light transport. *ACM Trans. Graph. 24*, 3 (2005).

[34] DUTRÉ, P., BALA, K., AND BEKAERT, P. *Advanced Global Illumination*. AK Peters, 2006.

[35] EDGERTON, H. E., AND KILLIAN, J. R. *Flash!: Seeing the unseen by ultra high-speed photography*. CT Branford Co., 1954.

[36] EINSTEIN, A. *Relativity: the special and the general theory*. Crown Publishers, 1961.

[37] FARO. Faro Technologies Inc.: Measuring Arms. http://www.faro.com, 2012.

[38] FERWERDA, J. A., SHIRLEY, P., PATTANAIK, S. N., AND GREENBERG, D. P. A model of visual masking for computer graphics. In *Proceedings of SIGGRAPH '97* (1997).

[39] FILIP, J., CHANTLER, M. J., GREEN, P. R., AND HAINDL, M. A psychophysically validated metric for bidirectional texture data reduction. *ACM Trans. Graph. 27*, 5 (2008).

[40] FILIP, J., CHANTLER, M. J., AND HAINDL, M. On uniform resampling and gaze analysis of bidirectional texture functions. *ACM Trans. Appl. Percept. 6*, 3 (2009).

[41] FILIP, J., AND HAINDL, M. Bidirectional texture function modeling: A state of the art survey. *IEEE Trans. on Pattern Analysis and Machine Intelligence 31* (2009).

[42] FLEMING, R. W., DROR, R. O., AND ADELSON, E. H. Real-world illumination and the perception of surface reflectance properties. *Journal of Vision 3*, 5 (2003), 347–368.

[43] FRANKLIN, J. Lorentz contraction, Bell's spaceships and rigid body motion in special relativity. *European Journal of Physics 31*, 2 (2010).

[44] FRIEDMAN, Y., AND GOFMAN, Y. A new relativistic kinematics of accelerated systems. *Physica Scripta 82*, 1 (2010).

[45] FUNKHOUSER, T., TSINGOS, N., AND JOT, J.-M. Survey of methods for modeling sound propagation in interactive virtual environment systems. *Presence and Teleoperation* (2003).

[46] GARCES, E., MARTIN, F., AND GUTIERREZ, D. Low cost decomposition of direct and global illumination in real scenes. In *Proceedings of CEIG'15* (2015).

[47] GBUR, G. A camera fast enough to watch light move? http://skullsinthestars.com/2012/01/04/a-camera-fast-enough-to-watch-light-move/, 2012.

[48] GELBART, A., REDMAN, B. C., LIGHT, R. S., SCHWARTZLOW, C. A., AND GRIFFIS, A. J. Flash lidar based on multiple-slit streak tube imaging lidar. vol. 4723, SPIE, pp. 9–18.

[49] GEORGIEV, I., KŘIVÁNEK, J., DAVIDOVIČ, T., AND SLUSALLEK, P. Light transport simulation with vertex connection and merging. *ACM Trans. Graph. 31*, 6 (2012).

[50] GEORGIEV, I., KŘIVÁNEK, J., HACHISUKA, T., NOWROUZEZAHRAI, D., AND JAROSZ, W. Joint importance sampling of low-order volumetric scattering. *ACM Trans. Graph. 32*, 6 (2013).

[51] GEORGIEV, T., AND LUMSDAINE, A. Theory and methods of lightfield photography. In *ACM SIGGRAPH ASIA '09 Courses* (2009), ACM.

[52] Gkioulekas, I., Levin, A., Durand, F., and Zickler, T. Micron-scale light transport decomposition using interferometry. *ACM Trans. Graph. 34*, 4 (2015).

[53] Goda, K., Tsia, K. K., and Jalali, B. Serial time-encoded amplified imaging for real-time observation of fast dynamic phenomena. *Nature 458* (2009), 1145–1149.

[54] Goldman, D. B., Gonterman, C., Curless, B., Salesin, D., and Seitz, S. M. Video object annotation, navigation, and composition. In *Proceedings of UIST '08* (2008), pp. 3–12.

[55] Gondek, J. S., Meyer, G. W., and Newman, J. G. Wavelength dependent reflectance functions. In *Proceedings of SIGGRAPH '94* (1994).

[56] Gortler, S. J., Grzeszczuk, R., Szeliski, R., and Cohen, M. F. The lumigraph. In *Proceedings of SIGGRAPH '96* (1996).

[57] Green, P., Sun, W., Matusik, W., and Durand, F. Multi-aperture photography. *ACM Trans. Graph. 26*, 3 (2007).

[58] Gupta, M., Nayar, S. K., Hullin, M., and Martin, J. Phasor imaging: A generalization of correlation-based time-of-flight imaging. *ACM Transactions on Graphics (TOG)* (2015).

[59] Gupta, O., Willwacher, T., Velten, A., Veeraraghavan, A., and Raskar, R. Reconstruction of hidden 3d shapes using diffuse reflections. *Optics Express 20*, 17 (2012).

[60] Guthe, M., Müller, G., Schneider, M., and Klein, R. BTF-CIELab: A perceptual difference measure for quality assessment and compression of BTFs. *Computer Graphics Forum 28*, 1 (2009).

[61] Gutierrez, D., Masia, B., and Jarabo, A. Computational photography. In *CEIG '12 Tutorials* (2012).

[62] Gutierrez, D., Muñoz, A., Anson, O., and Seron, F. Non-linear volume photon mapping. In *Proceedings of EGSR '05* (2005).

[63] Gutierrez, D., Narasimhan, S., Jensen, H., and Jarosz, W. Scattering. In *ACM SIGGRAPH Asia '08 Courses* (2008).

[64] Gutierrez, D., Seron, F., Muñoz, A., and Anson, O. Visualizing underwater ocean optics. *Computer Graphics Forum 27*, 2 (2008).

[65] Hachisuka, T., Jarosz, W., Georgiev, I., Kaplanyan, A., and Nowrouzezahrai, D. State of the art in photon density estimation. In *ACM SIGGRAPH ASIA '13 Courses* (2013).

[66] Hachisuka, T., Jarosz, W., and Jensen, H. W. A progressive error estimation framework for photon density estimation. *ACM Trans. Graph. 29*, 6 (2010).

[67] Hachisuka, T., and Jensen, H. W. Stochastic progressive photon mapping. *ACM Trans. Graph. 28*, 5 (2009).

[68] Hachisuka, T., Ogaki, S., and Jensen, H. W. Progressive photon mapping. *ACM Trans. Graph. 27*, 5 (2008).

[69] HaCohen, Y., Shechtman, E., Goldman, D. B., and Lischinski, D. Non-rigid dense correspondence with applications for image enhancement. *ACM Trans. Graph. 30*, 4 (2011).

[70] Haeberli, P. Paint by numbers: Abstract image representations. *SIGGRAPH Comput. Graph. 24*, 4 (1990), 207–214.

[71] Hamamatsu. Guide to Streak Cameras. http://sales.hamamatsu.com/assets/pdf/catsandguides/e_streakh.pdf, 2012.

[72] Han, C., Sun, B., Ramamoorthi, R., and Grinspun, E. Frequency domain normal map filtering. *ACM Trans. Graph. 26*, 3 (2007).

[73] Hanika, J., and Dachsbacher, C. Efficient monte carlo rendering with realistic lenses. *Computer Graphics Forum 33*, 2 (2014), 323–332.

[74] Hasinoff, S., Jozwiak, M., Durand, F., and Freeman, W. Search-and-replace editing for personal photo collections. In *Proceedings of ICCP '10* (2010), pp. 1–8.

[75] Havran, V., Filip, J., and Myszkowski, K. Bidirectional texture function compression based on the multilevel vector quantization. *Computer Graphics Forum 27*, 1 (2010).

[76] Hebden, J. Line scan acquisition for time-resolved imaging through scattering media. *Opt. Eng. 32*, 3 (1993), 626–633.

[77] Heer, J., and Bostock, M. Crowdsourcing graphical perception: using mechanical turk to assess visualization design. In *Proceedings of CHI '10* (2010).

[78] Heide, F., Hullin, M., Gregson, J., and Heidrich, W. Low-budget transient imaging using photonic mixer devices. *ACM Trans. Graph. 32*, 4 (2013).

[79] Heide, F., Xiao, L., Heidrich, W., and Hullin, M. B. Diffuse mirrors: 3D reconstruction from diffuse indirect illumination using inexpensive time-of-flight sensors. In *Proceedings of CVPR '15* (June 2014).

[80] Heide, F., Xiao, L., Kolb, A., Hullin, M. B., and Heidrich, W. Imaging in scattering media using correlation image sensors and sparse convolutional coding. *Opt. Express 22*, 21 (2014), 26338–26350.

[81] Hoaglin, D. C., and Iglewicz, B. Fine-tuning some resistant rules for outlier labeling. *Journal of the American Statistical Association 82*, 400 (1987), 1147–1149.

[82] Horn, D. R., and Chen, B. Lightshop: interactive light field manipulation and rendering. In *Proceedings of I3D '07* (2007), pp. 121–128.

[83] Hsiung, P.-K., and Dunn, R. H. P. Visualizing relativistic effects in spacetime. In *Proceedings of the Supercomputing Conference* (1989).

[84] Hsiung, P.-K., and Thibadeau, R. H. Spacetime visualization of relativistic effects. In *Proceedings of ACM Computer Science Conference '90* (February 1990).

[85] Hsiung, P.-K., Thibadeau, R. H., Cox, C. B., Dunn, R. H. P., Wu, M., and Olbrich, P. A. Wide-band relativistic Doppler effect visualization. In *Proceedings of Visualization '90* (1990), pp. 83–92.

[86] HSIUNG, P.-K., THIBADEAU, R. H., COXT, C. B., AND DUNN, R. H. P. Time dilation visualization in relativity. In *Proceedings of Supercomputing'90* (1990), pp. 835–844.

[87] HSIUNG, P.-K., THIBADEAU, R. H., AND WU, M. T-buffer: Fast visualization of relativistic effects in space-time. *SIGGRAPH Comput. Graph. 24,* 2 (1990).

[88] HU, X., DENG, Y., LIN, X., SUO, J., DAI, Q., BARSI, C., AND RASKAR, R. Robust and accurate transient light transport decomposition via convolutional sparse coding. *Optics Letters 39,* 11 (2014).

[89] HUANG, D., SWANSON, E., LIN, C., SCHUMAN, J., STINSON, W., CHANG, W., HEE, M., FLOTTE, T., GREGORY, K., AND PULIAFITO, C. Optical coherence tomography. *Science 254,* 5035 (1991), 1178–1181.

[90] IGLESIAS-GUITIAN, J. A., ALIAGA, C., JARABO, A., AND GUTIERREZ, D. A biophysically-based model of the optical properties of skin aging. *Computer Graphics Forum (EUROGRAPHICS 2015) 34,* 2 (2015).

[91] IHRKE, I., ZIEGLER, G., TEVS, A., THEOBALT, C., MAGNOR, M., AND SEIDEL, H.-P. Eikonal rendering: Efficient light transport in refractive objects. *ACM Trans. Graph. 26,* 3 (2007).

[92] ISAKSEN, A., MCMILLAN, L., AND GORTLER, S. J. Dynamically reparameterized light fields. In *Proceedings of SIGGRAPH '00* (2000), pp. 297–306.

[93] ITATANI, J., QUÉRÉ, F., YUDIN, G. L., IVANOV, M. Y., KRAUSZ, F., AND CORKUM, P. B. Attosecond streak camera. *Phys. Rev. Lett. 88* (2002), 173903.

[94] ITO, A., TAMBE, S., MITRA, K., SANKARANARAYANAN, A. C., AND VEERARAGHAVAN, A. Compressive epsilon photography for post-capture control in digital imaging. *ACM Trans. Graph. 33,* 4 (2014).

[95] JAKOB, W., AND MARSCHNER, S. Manifold exploration: A markov chain monte carlo technique for rendering scenes with difficult specular transport. *ACM Trans. Graph. 31,* 4 (2012).

[96] JARABO, A. Femto-photography: Visualizing light in motion. Master's thesis, Universidad de Zaragoza, 2012.

[97] JARABO, A., BUISAN, R., AND DIE. Bidirectional clustering for scalable VPL-based global illumination. In *Proceedings of CEIG '15* (2015).

[98] JARABO, A., MARCO, J., MUÑOZ, A., BUISAN, R., JAROSZ, W., AND GUTIERREZ, D. A framework for transient rendering. *ACM Trans. Graph. 33,* 6 (2014).

[99] JARABO, A., MASIA, B., BOUSSEAU, A., PELLACINI, F., AND GUTIERREZ, D. How do people edit light fields? *ACM Trans. Graph. 33,* 4 (2014).

[100] JARABO, A., MASIA, B., AND GUTIERREZ, D. Efficient propagation of light field edits. In *Proceedings of SIACG '11* (2011), pp. 75–80.

[101] JARABO, A., MASIA, B., AND GUTIERREZ, D. Transient rendering and relativistic visualization. Tech. Rep. TR-01-2013, Universidad de Zaragoza, April 2013.

[102] Jarabo, A., Masia, B., Velten, A., Barsi, C., Raskar, R., and Gutier-rez, D. Rendering relativistic effects in transient imaging. In *Proceedings of CEIG'13* (2013).

[103] Jarabo, A., Masia, B., Velten, A., Barsi, C., Raskar, R., and Gutier-rez, D. Relativistic effects for time-resolved light transport. *Computer Graphics Forum* (2015).

[104] Jarabo, A., Van Eyck, T., Sundstedt, V., Bala, K., Gutierrez, D., and O'Sullivan, C. Crowd light: Evaluating the perceived fidelity of illuminated dynamic scenes. *Computer Graphics Forum 31*, 2 (2012).

[105] Jarabo, A., Wu, H., Dorsey, J., Rushmeier, H., and Gutierrez, D. Effects of approximate filtering on the appearance of bidirectional texture functions. *IEEE Trans. on Visualization and Computer Graphics 20*, 6 (2014).

[106] Jarosz, W., Nowrouzezahrai, D., Sadeghi, I., and Jensen, H. W. A comprehensive theory of volumetric radiance estimation using photon points and beams. *ACM Trans. Graph. 30*, 1 (2011).

[107] Jarosz, W., Nowrouzezahrai, D., Thomas, R., Sloan, P.-P., and Zwicker, M. Progressive photon beams. *ACM Trans. Graph. 30*, 6 (2011).

[108] Jarosz, W., Schönefeld, V., Kobbelt, L., and Jensen, H. W. Theory, analysis and applications of 2D global illumination. *ACM Trans. Graph. 31*, 5 (2012).

[109] Jensen, H. W. *Realistic Image Synthesis Using Photon Mapping*. AK Peters, 2001.

[110] Jimenez, J., Zsolnai, K., Jarabo, A., Freude, C., Auzinger, T., Wu, X.-C., von der Pahlen, J., Wimmer, M., and Gutierrez, D. Separable subsurface scattering. *Computer Graphics Forum* (2015).

[111] Kadambi, A., Whyte, R., Bhandari, A., Streeter, L., Barsi, C., Dor-rington, A., and Raskar, R. Coded time of flight cameras: sparse deconvolution to address multipath interference and recover time profiles. *ACM Trans. Graph. 32*, 6 (2013).

[112] Kalli, H., and Cashwell, E. Evaluation of three Monte Carlo estimation schemes for flux at a point. Tech. Rep. LA-6865-MS, Los Alamos Scientific Lab, New Mexico, USA, 1977.

[113] Kaplanyan, A. S., and Dachsbacher, C. Adaptive progressive photon mapping. *ACM Trans. Graph. 32*, 2 (2013).

[114] Keller, M., and Kolb, A. Real-time simulation of time-of-flight sensors. *Simulation Modelling Practice and Theory 17*, 5 (2009).

[115] Kerr, W. B., and Pellacini, F. Toward evaluating lighting design interface paradigms for novice users. *ACM Trans. Graph. 28*, 3 (2009), 26:1–26:9.

[116] Kerr, W. B., and Pellacini, F. Toward evaluating material design interface paradigms for novice users. *ACM Trans. Graph. 29*, 4 (2010), 35:1–35:10.

[117] KIM, C., ZIMMER, H., PRITCH, Y., SORKINE-HORNUNG, A., AND GROSS, M. Scene reconstruction from high spatio-angular resolution light fields. *ACM Trans. Graph. 32*, 4 (2013).

[118] KIM, Y., WINNEMÖLLER, H., AND LEE, S. WYSIWYG stereo painting. In *Proceedings of I3D'13* (2013).

[119] KIRMANI, A., HUTCHISON, T., DAVIS, J., AND RASKAR, R. Looking around the corner using transient imaging. In *Proceedings of ICCV '09* (2009).

[120] KIRMANI, A., HUTCHISON, T., DAVIS, J., AND RASKAR, R. Looking around the corner using ultrafast transient imaging. *International Journal of Computer Vision 95*, 1 (2011), 13–28.

[121] KNAUS, C., AND ZWICKER, M. Progressive photon mapping: A probabilistic approach. *ACM Trans. Graph. 30*, 3 (2011).

[122] KODAMA, R., OKADA, K., AND KATO, Y. Development of a two-dimensional space-resolved high speed sampling camera. *Rev. Sci. Instrum. 70*, 625 (1999).

[123] KOLB, A., BARTH, E., KOCH, R., AND LARSEN, R. Time-of-flight sensors in computer graphics. *Computer Graphics Forum 29*, 1 (2010).

[124] KOLB, C., MITCHELL, D., AND HANRAHAN, P. A realistic camera model for computer graphics. In *Proceedings of the 22Nd Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1995), SIGGRAPH '95, ACM, pp. 317–324.

[125] KOPF, J., COHEN, M. F., LISCHINSKI, D., AND UYTTENDAELE, M. Joint bilateral upsampling. *ACM Trans. Graph. 26* (2007).

[126] KORTEMEYER, G., TAN, P., AND SCHIRRA, S. A slower speed of light: Developing intuition about special relativity with games. In *Proceedings of the International Conference on the Foundations of Digital Games (FDG'13)* (2013).

[127] KOUDELKA, M. L., MAGDA, S., BELHUMEUR, P. N., AND KRIEGMAN, D. J. Acquisition, compression, and synthesis of bidirectional texture functions. In *In ICCV 03 Workshop on Texture Analysis and Synthesis* (2003).

[128] KRAUS, U., RUDER, H., WEISKOPF, D., AND ZAHN, C. Was Einstein noch nicht sehen konnte - Visualisierung relativistischer Effekte. *Physik Journal 7/8* (2002).

[129] KŘIVÁNEK, J., GEORGIEV, I., HACHISUKA, T., VÉVODA, P., ŠIK, M., NOWROUZEZAHRAI, D., AND JAROSZ, W. Unifying points, beams, and paths in volumetric light transport simulation. *ACM Trans. Graph. 33*, 4 (2014).

[130] KULLA, C., AND FAJARDO, M. Importance sampling techniques for path tracing in participating media. *Computer Graphics Forum 31*, 4 (2012).

[131] KŘIVÁNEK, J., FERWERDA, J. A., AND BALA, K. Effects of global illumination approximations on material appearance. *ACM Trans. Graph. 29*, 4 (2010).

[132] Křivánek, J., Georgiev, I., Kaplanyan, A. S., and Cañada, J. Recent advances in light transport simulation: theory & practice. In *ACM SIGGRAPH '13 Courses* (2013).

[133] Lafortune, E. P., and Willems, Y. D. Bi-directional path tracing. In *Compugraphics '93* (1993).

[134] Latorre, P., Seron, F., and Gutierrez, D. Birefringency: Calculation of refracted ray paths in biaxial crystals. *The Visual Computer 28*, 4 (2012).

[135] Levin, A., Lischinski, D., and Weiss, Y. Colorization using optimization. *ACM Trans. Graph. 23* (August 2004), 689–694.

[136] Levoy, M., and Hanrahan, P. Light field rendering. In *Proceedings of SIGGRAPH '96* (1996).

[137] Lin, J., Liu, Y., Hullin, M. B., and Dai, Q. Fourier analysis on transient imaging with a multifrequency time-of-flight camera. In *Proceedings of CVPR '14* (2014).

[138] Lippmann, G. Épreuves réversibles donnant la sensation du relief. *Journal of Physics 7*, 4 (1908).

[139] Lippmann, G. La photographie integrále. *Academie des Sciences 146* (1908).

[140] Lischinski, D., Farbman, Z., Uyttendaele, M., and Szeliski, R. Interactive local adjustment of tonal values. *ACM Trans. Graph. 25* (2006), 646–653.

[141] Lo, W.-Y., van Baar, J., Knaus, C., Zwicker, M., and Gross, M. H. Stereoscopic 3d copy & paste. *ACM Trans. Graph. 29*, 6 (2010), 147.

[142] Luebke, D., Watson, B., Cohen, J. D., Reddy, M., and Varshney, A. *Level of Detail for 3D Graphics*. Elsevier Science Inc., New York, NY, USA, 2002.

[143] Lumière, L. The lumière cinematograph. *Journal of the Society of Motion Picture Engineers 27*, 6 (1936).

[144] Lumsdaine, A., and Georgiev, T. The focused plenoptic camera. In *Proceedings of ICCP '09* (2009).

[145] Lytro Inc. The Lytro camera. http://www.lytro.com, 2013.

[146] Ma, W.-C., Chao, S.-H., Tseng, Y.-T., Chuang, Y.-Y., Chang, C.-F., Chen, B.-Y., and Ouhyoung, M. Level-of-detail representation of bidirectional texture functions for real-time rendering. In *Proceedings of I3D '05* (2005).

[147] Marton, F., Agus, M., Gobbetti, E., Pintore, G., and Rodriguez, M. B. Natural exploration of 3d massive models on large-scale light field displays using the fox proximal navigation technique. *Computers & Graphics 36*, 8 (2012), 893 – 903.

[148] Masia, B., Jarabo, A., and Gutierrez, D. Favored workflows in light field editing. In *Proceedings of CGVCVIP '14* (2014).

[149] MASIA, B., WETZSTEIN, G., ALIAGA, C., RASKAR, R., AND GUTIERREZ, D. Display adaptive 3D content remapping. *Computers & Graphics* (2013).

[150] MASIA, B., WETZSTEIN, G., DIDYK, P., AND GUTIERREZ, D. A survey on computational displays: Pushing the boundaries of optics, computation and perception. *Computers & Graphics 37* (2013), 1012–1038.

[151] MATUSIK, W., PFISTER, H., BRAND, M., AND MCMILLAN, L. A data-driven reflectance model. *ACM Trans. Graph. 22*, 3 (2003).

[152] MCNAMARA, A., MANIA, K., AND GUTIERREZ, D. Perception in graphics, visualization, virtual environments and animation. In *ACM SIGGRAPH Asia '11 Courses* (2011).

[153] MESETH, J., MÜLLER, G., KLEIN, R., RÖDER, F., AND ARNOLD, M. Verification of rendering quality from measured btfs. In *Proceedings of APGV '06* (2006).

[154] MITCHELL, L. The man who stopped time. *Stanford Magazine* (2001).

[155] MITRA, K., AND KUMAR, S. Development and comparison of models for light-pulse transport through scattering-absorbing media. *Applied Optics 38*, 1 (1999).

[156] MÜLLER, G., MESETH, J., SATTLER, M., SARLETTE, R., AND KLEIN, R. Acquisition, synthesis and rendering of bidirectional texture functions. In *Eurographics '04 State of the Art Reports* (2004).

[157] MÜLLER, T., GROTTEL, S., AND WEISKOPF, D. Special relativistic visualization by local ray tracing. *IEEE Trans. on Visualization and Computer Graphics 16*, 6 (2010).

[158] MUSBACH, A., MEYER, G. W., REITICH, F., AND OH, S. H. Full wave modelling of light propagation and reflection. *Computer Graphics Forum 32*, 6 (2013).

[159] MYSZKOWSKI, K., TAWARA, T., AKAMINE, H., AND SEIDEL, H.-P. Perception-guided global illumination solution for animation rendering. In *Proceedings of SIGGRAPH '01* (New York, NY, USA, 2001), ACM.

[160] NAIK, N., BARSI, C., VELTEN, A., AND RASKAR, R. Estimating wide-angle, spatially varying reflectance using time-resolved inversion of backscattered light. *JOSA A 31*, 5 (2014).

[161] NAIK, N., KADAMBI, A., RHEMANN, C., IZADI, S., RASKAR, R., AND KANG, S. B. A light transport model for mitigating multipath interference in time-of-flight sensors. In *Proceedings of CVPR '15* (2015).

[162] NAIK, N., ZHAO, S., VELTEN, A., RASKAR, R., AND BALA, K. Single view reflectance capture using multiplexed scattering and time-of-flight imaging. *ACM Trans. Graph. 30* (2011).

[163] NAVARRO, F., SERON, F., AND GUTIERREZ, D. Motion blur rendering: State of the art. *Computer Graphics Forum 30*, 1 (2011).

[164] NAYAR, S., KRISHNAN, G., GROSSBERG, M. D., AND RASKAR, R. Fast Separation of Direct and Global Components of a Scene using High Frequency Illumination. *ACM Trans. Graph.* (Jul 2006).

[165] NG, R. Fourier slice photography. *ACM Trans. Graph. 24* (2005), 735–744.

[166] NOVÁK, J., NOWROUZEZAHRAI, D., DACHSBACHER, C., AND JAROSZ, W. Virtual ray lights for rendering scenes with participating media. *ACM Trans. Graph. 31, 4* (2012).

[167] O'DONOVAN, P., AGARWALA, A., AND HERTZMANN, A. Color compatibility from large datasets. *ACM Trans. Graph. 30, 4* (2011).

[168] O'TOOLE, M., HEIDE, F., XIAO, L., HULLIN, M. B., HEIDRICH, W., AND KUTULAKOS, K. N. Temporal frequency probing for 5D transient analysis of global light transport. *ACM Trans. Graph. 33, 4* (2014).

[169] O'TOOLE, M., AND KUTULAKOS, K. N. Optical computing for fast light transport analysis. *ACM Trans. Graph. 29, 6* (2010).

[170] OU, J., KARLÍK, O., KŘIVÁNEK, J., AND PELLACINI, F. Toward evaluating progressive rendering methods in appearance design tasks. *IEEE Comput. Graph. Appl. 33* (2012).

[171] PANDHARKAR, R., VELTEN, A., BARDAGJY, A., BAWENDI, M., AND RASKAR, R. Estimating motion and size of moving non-line-of-sight objects in cluttered environments. In *Proceedings of CVPR '11* (2011), pp. 265–272.

[172] PAULY, M., KOLLIG, T., AND KELLER, A. Metropolis light transport for participating media. In *Proceedings of EGWR '00* (2000).

[173] PEERS, P., MAHAJAN, D. K., LAMOND, B., GHOSH, A., MATUSIK, W., RAMAMOORTHI, R., AND DEBEVEC, P. Compressive light transport sensing. *ACM Trans. Graph. 28* (February 2009), 3:1–3:18.

[174] PELLACINI, F., AND LAWRENCE, J. Appwand: editing measured materials using appearance-driven optimization. *ACM Trans. Graph. 26* (2007).

[175] PERLIN, K. Improving noise. *ACM Trans. Graph. 21, 3* (2002).

[176] PHARR, M., AND HUMPHREYS, G. *Physically based rendering: From theory to implementation.* Morgan Kaufmann, 2010.

[177] POULI, T., CUNNINGHAM, D. W., AND REINHARD, E. A survey of image statistics relevant to computer graphics. *Computer Graphics Forum 30, 6* (2011).

[178] PRICE, B. L., AND COHEN, S. Stereocut: Consistent interactive object selection in stereo image pairs. In *Proceedings of ICCV '11* (2011), pp. 1148–1155.

[179] QI, H., REN, Y.-T., CHEN, Q., AND RUAN, L.-M. Fast method of retrieving the asymmetry factor and scattering albedo from the maximum time-resolved reflectance of participating media. *Applied Optics 54, 16* (2015).

[180] QIAO, H., LIN, J., LIU, Y., HULLIN, M. B., AND DAI, Q. Resolving transient time profile in tof imaging via log-sum sparse regularization. *Optics Letters 40, 6* (2015).

[181] Qu, J., Liu, L., Chen, D., Lin, Z., Xu, G., Guo, B., and Niu, H. Temporally and spectrally resolved sampling imaging with a specially designed streak camera. *Optics Letters 31* (2006), 368–370.

[182] Ramamoorthi, R., Mahajan, D., and Belhumeur, P. A first-order analysis of lighting, shading, and shadows. *ACM Trans. Graph. 26, 1* (2007).

[183] Ramanarayanan, G., Ferwerda, J., Walter, B., and Bala, K. Visual equivalence: towards a new standard for image fidelity. *ACM Trans. Graph. 26, 3* (2007).

[184] Rao, A. R., and Lohse, G. L. Towards a texture naming system: Identifying relevant dimensions of texture. *Vision Research 36, 11* (1996).

[185] Raskar, R. Computational photography: Epsilon to coded photography. In *Emerging Trends in Visual Computing*. Springer, 2009, pp. 238–253.

[186] Raskar, R., and Davis, J. 5D time-light transport matrix: What can we reason about scene properties? Tech. rep., MIT, 2008.

[187] Rief, H., Dubi, A., and Elperin, T. Track length estimation applied to point detector. *Nuclear Science and Engineering 87* (1984).

[188] Ritschel, T., Ihrke, M., Frisvad, J. R., Coppens, J., Myszkowski, K., and Seidel, H.-P. Temporal glare: Real-time dynamic simulation of the scattering in the human eye. *Computer Graphics Forum 28, 2* (2009).

[189] Rizzi, G., and Ruggiero, M. L. *Relativity in Rotating Frames*. Kluwer Academic, 2004.

[190] Rogowitz, B., and Rushmeier, H. Are image quality metrics adequate to evaluate the quality of geometric objects? In *Proceedings of SPIE '01, Vol 4299. Human Vision and Electronic Imaging VI* (2001).

[191] Rubinstein, M., Gutierrez, D., Sorkine, O., and Shamir, A. A comparative study of image retargeting. *ACM Trans. Graph. 29, 5* (2010).

[192] Ruiters, R., and Klein, R. BTF compression via sparse tensor decomposition. *Computer Graphics Forum 28, 4* (2009).

[193] Sadeghi, I., Muñoz, A., Laven, P., Jarosz, W., Seron, F., Gutierrez, D., and Jensen, H. W. Physically-based simulation of rainbows. *ACM Trans. Graph. 31, 1* (2012).

[194] Santosa, S., Chevalier, F., Balakrishnan, R., and Singh, K. Direct space-time trajectory control for visual media editing. In *Proceedings of CHI'13* (2013), pp. 1149–1158.

[195] Sattler, M., Sarlette, R., and Klein, R. Efficient and realistic visualization of cloth. In *Proceedings of Eurographics Symposium on Rendering 2003* (2003).

[196] Scott, D. W. *Multivariate Density Estimation: Theory, Practice, and Visualization*. Wiley, 1992.

[197] Seitz, S. M., and Kutulakos, K. N. Plenoptic image editing. *Int. Journal of Computer Vision 48, 2* (2002), 115–129.

[198] Seitz, S. M., Matsushita, Y., and Kutulakos, K. N. A theory of inverse light transport. In *in Proceedings of IEEE International Conference on Computer Vision* (2005), pp. 1440–1447.

[199] Sen, P., Chen, B., Garg, G., Marschner, S. R., Horowitz, M., Levoy, M., and Lensch, H. Dual Photography. *ACM Trans. Graph. 24*, 3 (2005), 745–755.

[200] Shiraga, H., Heya, M., Maegawa, O., Shimada, K., Kato, Y., Yamanaka, T., and Nakai, S. Laser-imploded core structure observed by using two-dimensional x-ray imaging with 10-ps temporal resolution. *Rev. Sci. Instrum. 66*, 1 (1995), 722–724.

[201] Shum, H.-Y., Sun, J., Yamazaki, S., Li, Y., and Tang, C.-K. Pop-up light field: An interactive image-based modeling and rendering system. *ACM Trans. Graph. 23*, 2 (2004), 143–162.

[202] Siltanen, S., Lokki, T., Kiminki, S., and Savioja, L. The room acoustic rendering equation. *J. Acoust. Soc. Am. 122*, 3 (2007).

[203] Silverman, B. W. *Density Estimation for Statistics and Data Analysis*. Taylor & Francis, 1986.

[204] Sloan, P.-P., Kautz, J., and Snyder, J. Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. *ACM Trans. Graph. 21*, 3 (2002).

[205] Smith, A., Skorupski, J., and Davis, J. Transient rendering. Tech. Rep. UCSC-SOE-08-26, School of Engineering, University of California, Santa Cruz, February 2008.

[206] Stavrakis, E., and Gelautz, M. Image-based stereoscopic painterly rendering. In *Proceedings of EGSR '04* (2004), pp. 53–60.

[207] Stokes, W. A., Ferwerda, J. A., Walter, B., and Greenberg, D. P. Perceptual illumination components: a new approach to efficient, high quality global illumination rendering. *ACM Trans. Graph. 23*, 3 (2004).

[208] Styer, D. F. How do two moving clocks fall out of sync? a tale of trucks, threads, and twins. *American Journal of Physics 75*, 9 (2007).

[209] Tamura, H., Mori, S., and Yamawaki, T. Textural features corresponding to visual perception. *IEEE Transactions on Systems, Man and Cybernetics 8*, 6 (1978).

[210] Tan, P., Lin, S., Quan, L., Guo, B., and Shum, H.-Y. Multiresolution reflectance filtering. In *Proceedings of Eurographics Symposium on Rendering 2005* (2005).

[211] Templin, K., Didyk, P., Ritschel, T., Myszkowski, K., and Seidel, H.-P. Highlight microdisparity for improved gloss depiction. *ACM Trans. Graph. 31*, 4 (2012).

[212] Terrell, J. Invisibility of the Lorentz contraction. *Phys. Rev. 116* (1959).

[213] Tompkin, J., Muff, S., Jakuschevskij, S., McCann, J., Kautz, J., Alexa, M., and Matusik, W. Interactive light field painting. In *ACM SIGGRAPH '12 Emerging Technologies* (2012), pp. 12:1–12:1.

[214] Tou, T. Y. Multislit streak camera investigation of plasma focus in the steady-state rundown phase. *IEEE Trans. Plasma Science 23* (1995), 870–873.

[215] Tsai, Y.-T., and Shih, Z.-C. K-clustered tensor approximation: A sparse multilinear model for real-time rendering. *ACM Trans. Graph. 31*, 3 (2012).

[216] Vaish, V., Wilburn, B., Joshi, N., and Levoy, M. Using plane + parallax for calibrating dense camera arrays. In *Proceedings of CVPR '04* (2004), pp. 2–9.

[217] Vangorp, P., Laurijssen, J., and Dutré, P. The influence of shape on the perception of material reflectance. *ACM Trans. Graph. 26*, 3 (2007).

[218] Veach, E. *Robust Monte Carlo methods for light transport simulation*. PhD thesis, Stanford, 1997.

[219] Veach, E., and Guibas, L. Bidirectional estimators for light transport. In *Proceedings of EGWR '94* (1994).

[220] Veach, E., and Guibas, L. J. Optimally combining sampling techniques for Monte Carlo rendering. In *Proceedings of SIGGRAPH '95* (1995).

[221] Veach, E., and Guibas, L. J. Metropolis light transport. In *Proceedings of SIGGRAPH '97* (1997).

[222] Velten, A., Fritz, A., Bawendi, M. G., and Raskar, R. Multibounce time-of-flight imaging for object reconstruction from indirect light. In *Conference for Lasers and Electro-Optics* (2012), OSA, p. CM2F.5.

[223] Velten, A., Willwacher, T., Gupta, O., Veeraraghavan, A., Bawendi, M. G., and Raskar, R. Recovering three-dimensional shape around a corner using ultrafast time-of-flight imaging. *Nature Communications*, 3 (2012).

[224] Velten, A., Wu, D., Jarabo, A., Masia, B., Barsi, C., Joshi, C., Lawson, E., Bawendi, M., Gutierrez, D., and Raskar, R. Femtophotography: Capturing and visualizing the propagation of light. *ACM Trans. Graph. 32*, 4 (2013).

[225] Velten, A., Wu, D., Jarabo, A., Masia, B., Barsi, C., Lawson, E., Joshi, C., Gutierrez, D., Bawendi, M. G., and Raskar, R. Relativistic ultrafast rendering using time-of-flight imaging. In *ACM SIGGRAPH '12 Talks* (2012).

[226] Velten, A., Wu, D., Masia, B., Jarabo, A., Barsi, C., Joshi, C., Lawson, E., Bawendi, M. G., Gutierrez, D., and Raskar, R. Transient imaging of macroscopic scenes at picosecond resolution. *Communications of ACM* (2015).

[227] Venkataraman, K., Lelescu, D., Duparré, J., McMahon, A., Molina, G., Chatterjee, P., Mullis, R., and Nayar, S. Picam: An ultra-thin high performance monolithic camera array. *ACM Trans. Graph. 32*, 6 (2013), 166:1–166:13.

[228] Vertical Horizon. Lightfield Iris. http://www.verticalhorizon-software.com/LFI/index.html, 2013.

[229] WANG, L., LIN, S., LEE, S., GUO, B., AND SHUM, H.-Y. Light field morphing using 2d features. *IEEE Trans. on Visualization and Computer Graphics 11*, 1 (2005), 25–34.

[230] WANNER, S., AND GOLDLUECKE, B. Globally consistent depth labeling of 4d light fields. In *Proceedings of CVPR '12* (2012), pp. 41–48.

[231] WATSON, A. B. Temporal sensitivity. *Handbook of perception and human performance 1* (1986).

[232] WEIDLICH, A., AND WILKIE, A. Realistic rendering of birefringency in uniaxial crystals. *ACM Trans. Graph. 27*, 1 (2008).

[233] WEISKOPF, D. Fast visualization of special relativistic effects on geometry and illumination. In *Data Visualization 2000*. 2000, pp. 219–228.

[234] WEISKOPF, D. An immersive virtual environment for special relativity. In *Proceedings of WSCG '00* (2000).

[235] WEISKOPF, D. *Visualization of Four-Dimensional Spacetimes*. PhD thesis, Eberhard-Karls-Universität Tübingen, 2001.

[236] WEISKOPF, D., BORCHERS, M., ERTL, T., FALK, M., FECHTIG, O., FRANK, R., GRAVE, F., KING, A., KRAUS, U., MULLER, T., NOLLERT, H.-P., RICA MENDEZ, I., RUDER, H., SCHAFHITZEL, T., SCHAR, S., ZAHN, C., AND ZATLOUKAL, M. Explanatory and illustrative visualization of special and general relativity. *IEEE Trans. on Visualization and Computer Graphics 12*, 4 (2006).

[237] WEISKOPF, D., KOBRAS, D., AND RUDER, H. Real-world relativity: Image-based special relativistic visualization. In *Proceedings of IEEE Visualization'00* (2000).

[238] WEISKOPF, D., KRAUS, U., AND RUDER, H. Searchlight and Doppler effects in the visualization of special relativity: A corrected derivation of the transformation of radiance. *ACM Trans. Graph. 18*, 3 (1999).

[239] WEISKOPF, D., KRAUS, U., AND RUDER, H. Illumination and acceleration in the visualization of special relativity: a comment on fast rendering of relativistic objects. *Journal of Visualization and Computer Animation 11*, 4 (2000).

[240] WESTIN, S. H., ARVO, J. R., AND TORRANCE, K. E. Predicting reflectance functions from complex surfaces. In *Proceedings of SIGGRAPH '92* (1992).

[241] WETZSTEIN, G., IHRKE, I., LANMAN, D., AND HEIDRICH, W. Computational plenoptic imaging. *Computer Graphics Forum 30*, 8 (2011).

[242] WETZSTEIN, G., LANMAN, D., HIRSCH, M., AND RASKAR, R. Tensor Displays: Compressive Light Field Synthesis using Multilayer Displays with Directional Backlighting. *ACM Trans. Graph. 31*, 4 (2012), 1–11.

[243] WILKIE, A., TOBLER, R. F., AND PURGATHOFER, W. Combined rendering of polarization and fluorescence effects. In *Proceedings of EGWR '01* (2001).

[244] WILLIAMS, L. Pyramidal parametrics. In *Proceedings of SIGGRAPH '83* (1983).

[245] Wu, D., Velten, A., O'Toole, M., Masia, B., Agrawal, A., Dai, Q., and Raskar, R. Decomposing global light transport using time of flight imaging. *International Journal of Computer Vision 105*, 3 (2013).

[246] Wu, D., Wetzstein, G., Barsi, C., Willwacher, T., Dai, Q., and Raskar, R. Ultra-fast lensless computational imaging through 5D frequency analysis of time-resolved light transport. *International Journal of Computer Vision* (2013).

[247] Wu, D., Wetzstein, G., Barsi, C., Willwacher, T., O'Toole, M., Naik, N., Dai, Q., Kutulakos, K., and Raskar, R. Frequency analysis of transient light transport with applications in bare sensor imaging. In *Proceedings of ECCV '12* (2012).

[248] Wu, H., Dorsey, J., and Rushmeier, H. Characteristic point maps. *Computer Graphics Forum 28*, 4 (2009).

[249] Wyant, J. C. White light interferometry. In *Proceedings of SPIE '02* (2002), vol. 4737, pp. 98–107.

[250] Xia, H., and Zhang, C. Ultrafast ranging lidar based on real-time Fourier transformation. *Optics Letters 34* (2009), 2108–2110.

[251] Xu, K., Li, Y., Ju, T., Hu, S.-M., and Liu, T.-Q. Efficient affinity-based edit propagation using kd tree. *ACM Trans. Graph. 5*, 28 (2009).

[252] Xu, L., Yan, Q., and Jia, J. A sparse control model for image and video editing. *ACM Trans. Graph. 32*, 6 (2013).

[253] Yoo, K. M., and Alfano, R. R. Time-resolved coherent and incoherent components of forward light scattering in random media. *Optics Letters 15*, 6 (1990).

[254] Yücer, K., Jacobson, A., Hornung, A., and Sorkine, O. Transfusive image manipulation. *ACM Trans. Graph. 31*, 6 (2012), 176:1–176:9.

[255] Žalik, K. R. An efficient k-means clustering algorithm. *Pattern Recognition Letters 29*, 9 (2008), 1385–1391.

[256] Zell, E., Aliaga, C., Jarabo, A., Zibrek, K., Gutierrez, D., McDonnell, R., and Botsch, M. To stylize or not to stylize? effect of shape and material stylization on the perception of computer generated faces. *ACM Trans. Graph. 34*, 6 (2015).

[257] Zhang, Y., Yi, H., and Tan, H. One-dimensional transient radiative transfer by lattice Boltzmann method. *Optics Express 21*, 21 (2013).

[258] Zhang, Z., Wang, L., Guo, B., and Shum, H.-Y. Feature-based light field morphing. *ACM Trans. Graph. 21*, 3 (2002).

[259] Zhu, C., and Liu, Q. Review of Monte Carlo modeling of light transport in tissues. *J. Biomed. Opt. 18*, 5 (2013).

[260] Zwicker, M., Jarosz, W., Lehtinen, J., Moon, B., Ramamoorthi, R., Rousselle, F., Sen, P., Soler, C., and Yoon, S.-E. Recent advances in adaptive sampling and reconstruction for monte carlo rendering. *Computer Graphics Forum (Proceedings of Eurographics) 34*, 2 (May 2015), 667–681.