Linköping Studies in Science and Technology
Dissertation No. 2046

Computational Photography: High Dynamic Range and Light Fields

Saghi Hajisharif

li.u LINKÖPING
UNIVERSITY

Linköping studies in science and technology.
Dissertation, No. 2046

COMPUTATIONAL PHOTOGRAPHY: HIGH DYNAMIC RANGE
AND LIGHT FIELDS

Saghi Hajisharif

Division of Media and Information Technology
Department of Science and Technology
Linköping University, SE-601 74 Norrköping, Sweden
Norrköping, February 2020

Description of the cover image

The cover of this thesis represents computational photography for light field imaging and HDR imaging. The light manifold illustrates the light field and the RGB band is a combination of different color filter arrays (CFAs) with spatially interlaced multi-exposure patterns that have been used for the HDR imaging research presented in this thesis.

Computational Photography: High Dynamic Range and Light Fields

Copyright © 2020 Saghi Hajisharif (unless otherwise noted)

*Division of Media and Information Technology
Department of Science and Technology
Linköping University, Campus Norrköping
SE-601 74 Norrköping, Sweden*

ISBN: 978-91-7929-905-7

ISSN: 0345-7524

Printed in Sweden by LiU-Tryck, Linköping, 2020

Abstract

The introduction and recent advancements of computational photography have revolutionized the imaging industry. Computational photography is a combination of imaging techniques at the intersection of various fields such as optics, computer vision, and computer graphics. These methods enhance the capabilities of traditional digital photography by applying computational techniques both during and after the capturing process. This thesis targets two major subjects in this field: *High Dynamic Range* (HDR) image reconstruction and *Light Field* (LF) compressive capturing, compression, and real-time rendering.

The first part of the thesis focuses on the HDR images that concurrently contain detailed information from the very dark shadows to the brightest areas in the scenes. One of the main contributions presented in this thesis is the development of a unified reconstruction algorithm for spatially variant exposures in a single image. This method is based on a camera noise model, and it simultaneously resamples, reconstructs, denoises, and demosaics the image while extending its dynamic range. Furthermore, the HDR reconstruction algorithm is extended to adapt to the local features of the image, as well as the noise statistics, to preserve the high-frequency edges during reconstruction.

In the second part of this thesis, the research focus shifts to the acquisition, encoding, reconstruction, and rendering of light field images and videos in a real-time setting. Unlike traditional integral photography, a light field captures the information of the dynamic environment from all angles, all points in space, and all spectral wavelength and time. This thesis employs sparse representation to provide an end-to-end solution to the problem of encoding, real-time reconstruction, and rendering of high dimensional light field video data sets. These solutions are applied on various types of data sets, such as light fields captured with multi-camera systems or hand-held cameras equipped with micro-lens arrays, and spherical light fields. Finally, sparse representation of light fields was utilized for developing a single sensor light field video camera equipped with a color-coded mask. A new compressive sensing model is presented that is suitable for dynamic scenes with temporal coherency and is capable of reconstructing high-resolution light field videos.

Populärvetenskaplig Sammanfattning

Beräkningsfotografi (från engelskans *computational photography*) kombinerar optik, bildsensorer och beräkning för att utöka eller skapa helt nya möjligheter för kamerabaserad avbildning eller mätning av både helt vardagliga eller mycket specifika scener eller objekt. Forskningsområdet som ligger i skärningspunkten mellan forskningsområdena optik, datorseende och datorgrafik har under det senaste decenniet genererat ett antal nya tillämpningar och forskningsfrågeställningar som kommer att forma hur vi i framtiden avbildar en scen med en kamera. Två av dessa, vilka också ligger i fokus för den här avhandlingen, är rekonstruktion av bilder med hög dynamiskt omfång (HDR, från engelskans *High Dynamic Range*) samt komprimering och syntetisering (rendering) av ljusfält (från engelskans *light field*) för både still- och rörlig bild.

Bilder med högt dynamiskt omfång kan användas inom många olika områden, såsom filminspelning, datorseende och rendering av syntetiska bilder med fotorealistisk ljussättning för att nämna några. De innehåller detaljerad information om både mörka och ljusa områden i scener samtidigt, något som bilder tagna med en vanlig kamera inte kan. Tekniken för avbildning i HDR har de senaste åren mognat, och används idag i nästan alla mobiltelefonkameror för att förbättra bildkvaliteten. Den första delen av avhandlingen presenterar algoritmer för att rekonstruera HDR-bilder från multipla exponeringar av en bild. Dessa metoder, vilka är baserade på modeller av sensors brusegenskaper, rekonstruerar och reducerar brus i bilden samtidigt som den utökar dess dynamiska omfång.

Den andra delen av avhandlingen handlar om att fånga in och rekonstruera ljusfälts-bilder och video i realtid. Till skillnad från traditionell fotografering fångar ljusfält informationen om miljön från alla vinklar och punkter i rummet, inte bara i 2D. Den här informationen kan sedan användas, till exempel för att ändra fokus i den tagna bilden eller för skattning av avstånd eller rekonstruktion av 3D-modeller med hjälp av datorseende-algoritmer. Ytterligare exempel är att använda den här informationen för 3D-visualisering av infångade objekt och miljöer, till exempel för tillämpning inom virtuell verklighet (VR). Ljusfält är mycket hög-dimensionell data, vilket innebär utmaningar för insamling, lagring och visualisering av datan. Den här avhandlingen presenterar en rad algoritmer och metoder som löser dessa utmaningar dels genom att reducera datamängden vid insamling och lagring, samt genom effektiva rekonstruktionsalgoritmer specifikt designade för visualisering av stora mängder visuell data i realtid.

*“For my part I know nothing with
any certainty, but the sight of the
stars makes me dream.”*

Vincent Van Gogh

Acknowledgments

During my time as a researcher and Ph.D. student, I had the privilege to get support from and work with many amazing people. First, and for most, I would like to express my sincere gratitude to my supervisor Professor *Jonas Unger*, who was a constant source of encouragement throughout my graduate career. Jonas introduced me to the fantastic world of HDR imaging and later on Light field photography. Thank you, Jonas, for your enduring patience and going above and beyond by making sure the paper drafts are adequately edited and finalized for the submissions. I would like to thank my co-supervisor, Professor *Anders Ynnerman*, for providing a unique research atmosphere at the division of *Media and Information Technology* and giving me a chance to be a part of it.

I would like to acknowledge the excellent work of our collaborators and co-authors who have contributed to the research presented in this thesis: *Joel Kronander*, *Ehsan Miandji*, *Per Larsson*, *Kiet Tran*, *Christine Guillemot*, and *Gabriel Baravdish*.

A lot of things would not be possible without the help of our wonderful administrator *Eva Skärblom*! I am grateful to you for all your help and kind heart!

I would like to thank the *Institute for Creative Technologies (ICT)* for hosting my research visit at the Graphics Lab with the supervision of *Paul Debevec*. I am grateful to *Andrew Jones*, *Koki Nagano*, and *Jay Busch* for teaching me about the light-stage and automultiscopic display. Above that, for making it a memorable stay with all the fun in L.A.. I would like to also thank *Zahra Nazari* for her endless support during my research visit at ICT.

The highlight of my research was the daily interactions with my current and former colleagues at the division of *Media and Information Technology* at the *Science and Information Technology (ITN)* department at Linöping University. My sincere thanks go to my amazing colleagues and friends at Visual Computing Laboratory (VCL). I can think of no finer individual than *Apostolia Tsirikoglou*, my wonderful office mate, and a true friend both in and out of the lab. You made my days better with your presence and unconditional support. I am grateful to *Ehsan Miandji*, who introduced me to the fantastic world of compression and compressive sensing, which led to the light field imaging project. Your passion for science and encouragement motivated me every day! I am grateful to *Gabriel Eilertsen* for helping me with my HDR-related questions, exciting discussions about deep learning methods, organizing paper reading sessions, and many more. I enjoyed all our interesting talks in and outside of the lab! I want to thank *Tanaboon Tongbuasirilai* for bringing enthusiasm about BRDF to the group, and assuring me that the future will be just fine. I had the privilege to work with *Joel Kronander*

on the HDR imaging project and learned a lot from his passion for this field. Nothing would have been possible without *Per Larsson* who was behind designing and implementing our capturing devices for light fields and HDR. I would also like to extend my thanks to my former and current colleagues at VCL: *Andrew Gardner, Erik Olsson, Gabriel Baravdish, Karin Stacke* and *Milda Pocevičiute*. Every day spent with you was a fun day of research, and I enjoyed our interesting conversations! It was great to be working with you guys!

This thesis has been proofread by the following people: *Jonas Unger, Ehsan Miandji, Apostolia Tsirikoglou, Gabriel Eilertsen, Indre Genelyte* and *Gabriel Baravdish*. I appreciate your insightful comments that helped to improve the text.

A PhD life is not possible without supportive and loving friends, who helped me with keeping my sanity with fun activities outside of the lab; My heartfelt thanks to all my wonderful friends, specially: *Apostolia, Indre, Alex, Mina, Babak, Anna, Hamid, Parisa, Ali, Negar, Arash, Niki, Andrew, Sherielyn, Navid, Sepideh*, and *Naghmeh*.

I would like to express my gratitude to my dear parents *Nassrin* and *Hassan* for inspiring me to push the boundaries and move forward. Your countless sacrifices made it possible for me to pursue my dreams! Thank you *Sohrab*, your patience with my maths and physics questions finally paid off! And my dear *Arezo*, thanks for always being there for me. *Maman Mahin*, I am grateful to you for your sleepless nights whenever I had an exam! Special thanks to my second parents *Shamsi* and *Davood* for their loving supports over the past year.

Last but not least, my deepest gratitude goes to my dearest friend, colleague, and husband, *Ehsan Miandji*, who has been the main pillar of support since I started my research. You have been the light that brightens up my days. You encouraged me when the research did not go well, and I was in doubt. You celebrated each success with me and showed me the true value of love. This dissertation would not have been possible without you!

Saghi Hajisharif
Norrköping, January 2020

List of Publications

This thesis includes research that is based the publications listed below:

- S. Hajisharif, J. Kronander, and J. Unger, “HDR Reconstruction for Alternating Gain (ISO) Sensor Readout,” in *Eurographics 2014 - Short Papers*, E. Galin and M. Wand, Eds. The Eurographics Association, 2014
- S. Hajisharif, J. Kronander, and J. Unger, “Adaptive DualISO HDR Reconstruction,” *EURASIP Journal on Image and Video Processing*, vol. 2015, no. 41, dec 2015
- E. Miandji, S. Hajisharif, and J. Unger, “A Unified Framework for Compression and Compressed Sensing of Light Fields and Light Field Videos,” *ACM Trans. Graph.*, vol. 38, no. 3, pp. 23:1–23:18, may 2019
- S. Hajisharif, E. Miandji, P. Larsson, K. Tran, and J. Unger, “Light field video compression and real time rendering,” *Computer Graphics Forum*, vol. 38, no. 7, pp. 265–276, 2019
- S. Hajisharif, E. Miandji, G. Baravadish, and J. Unger, “Compression and Real-Time Rendering of Inward Looking Spherical Light Fields,” in *Eurographics 2020 - Short Papers*, U. Assarsson and D. Panozzo, Eds. The Eurographics Association, (Submitted)
- S. Hajisharif, E. Miandji, C. Guillemot, and J. Unger, “Single Sensor Compressive Light Field Video Camera,” in *Eurographics 2020*, U. Assarsson and D. Panozzo, Eds. The Eurographics Association, (Conditionally Accepted)

Other publications by the author that are relevant to this thesis but are not included are a conference paper on HDR image based lighting and a chapter of a book:

- S. Hajisharif, J. Kronander, E. Miandji, and J. Unger, “Real-time Image Based Lighting with Streaming HDR Light Probe Sequences,” in *SIGRAD 2012*. Sweden: Linköping University Electronic Press, 2012
- J. Unger, J. Kronander, and S. Hajisharif, “Unified Reconstruction of Raw HDR Video Data,” in *High Dynamic Range Video: From Acquisition, to Display and Applications*, 1st ed., F. Dufaux, P. L. Callet, R. K. Mantiuk, and M. Mrak, Eds. Elsevier, 2016, ch. 2, pp. 62–83

Contributions

The contributions made in this thesis are in the area of computational photography ranging from *High Dynamic Range* (HDR) reconstruction to *Light Field* (LF) imaging. The thesis is divided into two main components: HDR and LFs. The HDR reconstruction is the main focus of the first part of the thesis (*Paper A* and *Paper B*), while LF acquisition, compression, and reconstruction is the centerpiece of the second part (*Paper C*, *Paper D*, *Paper E*, and *Paper F*). In what follows, the publications included in this thesis are listed with a short description and the author’s contributions.

Paper A: HDR Reconstruction for Alternating Gain (ISO) Sensor Read-out

S. Hajisharif, J. Kronander, and J. Unger, “HDR Reconstruction for Alternating Gain (ISO) Sensor Readout,” in *Eurographics 2014 - Short Papers*, E. Galin and M. Wand, Eds. The Eurographics Association, 2014

The main idea comes from a unified framework for the reconstruction of HDR video from multiple exposure images [9]. The framework was modified and applied on a spatially multiplexed image captured with multiple gain settings on a conventional camera which extended the dynamic range of the captured image by 2 – 3 f-stops. The result was presented at Eurographics 2014 as a short paper by the author of this thesis. The author was the main contributor behind the design and implementation of the method as well as capturing and providing the data and the majority of the written manuscript. This work has been also published in a book chapter [8].

Paper B: Adaptive DualISO HDR Reconstruction

S. Hajisharif, J. Kronander, and J. Unger, “Adaptive DualISO HDR Reconstruction,” *EURASIP Journal on Image and Video Processing*, vol. 2015, no. 41, dec 2015

This paper presents a novel HDR reconstruction where the size of the filter kernel adjusts to the statistical features of the camera noise model and the image structure, preserving the edges and important features of the scene. This work is a continuation of the work that is presented in *Paper A*. The result was published as a journal article and the author was responsible for the design and the implementation of the method as well as its written presentation.

Paper C: A Unified Framework for Compression and Compressed Sensing of Light Fields and Light Field Videos

E. Miandji, S. Hajisharif, and J. Unger, “A Unified Framework for Compression and Compressed Sensing of Light Fields and Light Field Videos,” *ACM Trans. Graph.*, vol. 38, no. 3, pp. 23:1–23:18, may 2019

The paper presents a novel compressed sensing and compression framework for multidimensional visual data. Two applications are presented for light field images and light field videos. The author of this thesis has made close contributions on the developments of the framework and the design of the real-time rendering and reconstruction of light fields and light field videos. The author also assisted in writing and editing the manuscript and providing results in terms of images and videos.

Paper D: Light Field Video Compression and Real-Time Rendering

S. Hajisharif, E. Miandji, P. Larsson, K. Tran, and J. Unger, “Light field video compression and real time rendering,” *Computer Graphics Forum*, vol. 38, no. 7, pp. 265–276, 2019

The paper presents a framework for compression and real-time rendering of light field videos. This is an extension of *Paper C* where an in-depth study of the effect of image noise on the compression efficiency is carried out. Additionally, a novel method is proposed for improving the efficiency of the compression by pruning the trained ensemble of dictionaries. Real-time reconstruction and novel view generation of the compressed data has also been implemented. An application for this framework was also presented for heart surgery documentation using light field videos. The author of this thesis was the main contributor to the paper that closely collaborated with E. Miandji and J. Unger and contributed to the majority of the implementations and written presentation. The results of this project was presented at the *Pacific Graphics 2019* conference by the author.

Paper E: Compression and Real-time Rendering of Inward Spherical Light Fields

S. Hajisharif, E. Miandji, G. Baravadish, and J. Unger, “Compression and Real-Time Rendering of Inward Looking Spherical Light Fields,” in *Eurographics 2020 - Short Papers*, U. Assarsson and D. Panozzo, Eds. The Eurographics Association, (Submitted)

As a continuation of the previous light field compression and rendering methods in *Paper D* and *Paper E*, we applied a similar method to a new light field format, where the cameras are mounted on a moving arm rotating and capturing the object at the center. We explored the effect of entropy coding of the sparse coefficients on the compression ratio and reconstruction quality. The author of this thesis was the main contributor of this project as well as the written manuscript.

Paper F: Single Sensor Compressive Light Field Video

S. Hajisharif, E. Miandji, C. Guillemot, and J. Unger, “Single Sensor Compressive Light Field Video Camera,” in *Eurographics 2020*, U. Assarsson and D. Panozzo, Eds. The Eurographics Association, (Conditionally Accepted)

This work is based on the idea of using compressed sensing for capturing high dimensional visual data. The paper proposes a novel reconstruction solution for a camera design where a mask is placed between the aperture and the sensor. The proposed reconstruction algorithm takes advantage of the correlations in the consecutive frames of a light field video to enable accurate reconstruction of the data in the temporal as well as spatial and angular domains. The author is responsible for the idea, design, and implementation of the method in addition to the written presentation. This paper is submitted to Eurographics 2020, and conditionally accepted.

Contents

Abstract	v
Populärvetenskaplig Sammanfattning	vii
Acknowledgments	xi
List of Publications	xiii
Contributions	xv
1 Introduction	1
1.1 Digital Imaging	2
1.1.1 High Dynamic Range Imaging	4
1.1.2 Light Field Imaging	6
1.1.3 Applications	7
1.2 Objectives and Contributions	9
1.3 Thesis Outline	11
2 Fundamental Camera Concepts	13
2.1 In-Camera Processing	13
2.1.1 Color Filter Array	14
2.1.2 Camera Parameters	15
2.1.3 Camera Noise Sources	16
2.2 Dynamic Range	18
2.3 Quality Metrics	19
2.3.1 PSNR	19
2.3.2 SSIM	20
2.3.3 HDR Quality Metrics	20
3 High Dynamic Range Imaging	23
3.1 Outline and Contributions	24
3.2 HDR Acquisition Overview	24
3.2.1 Multi Shot - Single Sensor	24
3.2.2 Single Shot - Multiple Sensors	26
3.2.3 Single Shot - Single Sensor	27
3.3 HDR Reconstruction Overview	28
3.4 Dual-ISO Spatial Multiplexing	31
3.4.1 Sensor Noise Model	31
3.4.2 Variance Estimate	32
3.4.3 Camera Parameters Calibration	33

3.4.4	Local Polynomial Approximation	34
3.4.5	Maximum Localized Likelihood Fitting	35
3.4.6	Adaptive Kernel Regression	37
3.4.7	Update Rule 1: Error of Estimation Versus Standard Deviation (EVS)	38
3.4.8	Update Rule 2: Intersection of Confidence Intervals (ICI)	41
3.4.9	Spatially Interlaced Gain Patterns	42
3.5	Summary and Future Work	43
4	Sparse Representation and Compressed Sensing	45
4.1	Sparse Signal Representation	45
4.2	Sparse Dictionary Learning	47
4.2.1	Patch-Based Learning	48
4.2.2	Examples of Dictionary Training Algorithms	48
4.3	Compressive Sensing	49
5	Light Field Compression and Rendering	53
5.1	Outline and Contributions	53
5.2	Plenoptic Function	55
5.3	Light Field Data Points	56
5.4	Light Field Compression and Rendering	56
5.4.1	Multidimensional Dictionary Ensemble	58
5.5	Aggregated Multidimensional Dictionary Ensemble	60
5.5.1	Pre-Clustering	61
5.5.2	AMDE Encoding	64
5.5.3	AMDE Decoding	65
5.5.4	Dictionary Ensemble Pruning	65
5.5.5	Denoising Prior to Compression	67
5.5.6	Real-Time Decoding and Rendering of Light Field Video	69
5.5.7	Quantization and Entropy Coding	72
5.6	Inward Looking Spherical Light Fields	73
5.7	Summary and Future Work	74
6	Compressive Light Field Imaging	77
6.1	Outline and Contributions	77
6.2	Light Field Acquisition Techniques	78
6.2.1	Multi Sensor LF Acquisition	79
6.2.2	Single Sensor LFs	80
6.3	Compressive Light Field Video Camera	82
6.3.1	Sensing Matrix Design	84
6.3.2	Dictionary Training	85
6.3.3	Sparse Reconstruction	86

6.4	Summary and Future Work	90
7	Concluding Remarks and Outlook	93
7.1	Summary of Contributions	93
7.2	Future Work	94
	Bibliography	99
	Publications	125
	Paper A	127
	Paper B	135
	Paper C	151
	Paper D	179
	Paper E	195
	Paper F	203

Introduction

Digital cameras have been an omnipresent technology, from mobile phones and DSLR cameras to security systems and industrial quality control applications. Traditional imaging relies mainly on hardware systems for capturing a scene. Although the hardware of digital cameras has evolved very fast during the past decades, they are still physically limited in many aspects, particularly the sensor's capabilities to capture challenging scenes. *Computational photography* is an emerging field that intersects with computer vision, computer graphics, and applied optics, and provides solutions to the limitations of digital cameras through new optical designs, and mathematical and computational techniques. These algorithms have already been incorporated into many digital camera software systems to improve the capturing process and assist the user. As an example, face detection is commonly used to identify people in a scene and adjust the focus, exposure, and camera settings to obtain the best image of each person. De-blurring and denoising algorithms have long been parts of the capturing system. Algorithms for capturing higher dynamic range are now available in most consumer cameras, where typically multiple shots are captured and fused to improve the final image. Tone mapping is used to map this high dynamic range image to the dynamic range of the display used to view the image.

Plenoptic imaging is a subset of computational photography aiming to acquire all or some of the dimensions of the plenoptic function, such as the spatial, angular, spectral, and temporal domains using both hardware modulations and computational algorithms. These algorithms can be utilized in various applications from medical imaging [10], saliency detection [11], scene flow estimation [12], increasing

Figure 1.1: *Digital camera pipeline: Camera optics, color filter array (CFA), sensor, gain control, analogue-to-digital (A/D) converter, camera processing unit consisting of different stages to covert the raw digital input to an 8-bit final image. The green box indicates the analog signal and the blue box contains the processes that are applied on the digital signal.*

the dynamic range in photography [13], image denoising [14], super-resolution for zooming in handheld devices [15], wide aperture effects to synthesize shallow depth of field [16], and many more. The purpose of this thesis is to improve the existing algorithms and extend the field of computational photography in the direction of high dynamic range (HDR) and light field (LF) imaging.

This chapter provides a brief introduction to the topics of computational photography and focuses on plenoptic imaging, which is most relevant to the research conducted in this thesis. It starts with an introduction to traditional digital imaging, HDR imaging, and light field imaging, and concludes with the objectives of the thesis and a list of contributions by the author of this thesis towards these objectives.

1.1 Digital Imaging

The traditional digital imaging pipeline, as shown in Figure 1.1, consists of an optical unit such as lens and aperture, color filter array (CFA), a sensor, gain amplifier, analog-to-digital (A/D) converter, and a digital image processing unit. The incoming luminance passes through the optics of the camera and the CFA and is projected onto the sensor. The sensor converts the incoming photons to

electric charges and stores them in the photosites or capacitors while the shutter is open. Too much light might lead to the overflow of the pixel capacitors and saturate the image at the corresponding pixels. The gain control unit, which accommodates the ISO settings on modern cameras, amplifies the analog signal before it is quantized into the digital signal. Amplification of the analog signal is necessary to bring the voltage to a range that is required to get a desired digital output in the A/D converter stage. In the digital image processing unit, the raw digital pixel values are adjusted for the black level. The black level is usually calculated from a dark frame in the raw data measuring the sensor noise. The white balancing step transfers the image to a new space to mimic the chromatic adaptation of the eye so that the colors are represented correctly. Additionally, the image is debayered (or demosaiced) [17] to reconstruct all color channels for each pixel. Finally, the camera response function curve is applied on the color image along with other enhancement algorithms, and the image is compressed to an 8-bit JPEG format.

One of the main shortcomings of the traditional 2D imaging is the constraint of capturing a single point in space at a given time. It is prevalent that the artists and directors would like to change the focus point or the camera's viewpoint to make a scene more attractive according to artistic intentions. Usually, they are required to re-shoot the scene, which is time-consuming, costly, and in some cases, not possible. Another limitation is the incapability of sensors to capture all the luminance coming from the scene, limiting the dynamic range of the acquired image. As an example, it is incredibly challenging to capture an indoor scene simultaneously with what is seen through a window.

In order to extend the capabilities of digital imaging, new optical designs, new sensors, sampling methods, and algorithms are required to handle data and re-create the real scene as close to the original as possible. Computational photography provides the platform for developing optical designs and algorithms for recovering the original signal and transforming it into a proper format for displaying on a suitable device. The optical design modifies the camera by adding optical elements such as ND-filters [18], beam-splitter [19, 20], microlens arrays [21], and color-masks [22, 23] to the optical path of the camera to enhance the acquisition process. Figure 1.2 illustrates a simplified pipeline for computational photography. Please note that the elements showed in this figure, such as microlens array and spatially interlaced exposures, are not necessarily built together. The raw data from A/D converter is passed to a unit consisting of different computational methods that reconstruct an estimate of the original signal. Unlike the traditional camera pipeline that compresses the image using JPEG format, computational methods find the proper encoder and decoder based on the context of the image, the display technology, and the application at hand.

Figure 1.2: A simplified pipeline for computational photography. The optical designs modify the camera architecture by adding optical elements such as ND-filters, microlens arrays, and modifying the gain control per pixel as shown in the optical and hardware modification unit (purple box). The CFA pattern can be adapted to the features of the scene, and the gain settings can be controlled to enable multi-exposure capturing. The digital raw data is then processed for a specific task and then compressed for storage. To display the captured data, the image is reconstructed and visualized on a suitable device.

By designing suitable algorithms, computational photography enables us to process the captured data efficiently and recover as much information as possible for a given optical design. This thesis contributes to the development of these algorithms for retrieving and storing high dimensional information such as light fields from a real environment. The following sections will explain a few examples of computational photography, which is most related to the topics covered by this thesis.

1.1.1 High Dynamic Range Imaging

One of the areas in which digital cameras rely heavily on computational methods is capturing a scene with a high dynamic range, or enhancing the dynamic range of the acquired image in a post-processing phase. high dynamic range imaging is a technique used to digitally capture a wide range of luminance in the scene, including the visible light from direct sunlight to the darkest shadows. The actual luminance values of the world can be measured in an HDR image and used to illuminate 3D objects through a family of techniques known as Image Based Lighting (IBL) [24]. IBL is used in applications such as photo-realistic rendering, relighting for virtual and augmented reality, entertainment, cinematography and gaming industry [7, 25, 26, 27, 28, 29]. There are many other applications of HDR imaging, including, but not limited to, astrophotography [30, 31], medical imaging [32, 33, 34, 35], and display systems [36, 37].

Figure 1.3: (a) *"The Great Wave"*, photograph by Gustave Le Gray, 1857, Albumen print from collodion-on-glass negative. Museum no. 68:004, ©Victoria and Albert Museum, London. (b) Wyckoff's image of *"Ivy Mike"*, the first hydrogen bomb explosion, appeared on the cover of *Life* magazine on April 19, 1954.

What we describe as HDR photography today, was first introduced by Gustave Le Gray in 1850, who combined two negatives into a single positive print to capture the extreme dynamic range of the sky and the sea, Figure 1.3 (a). The principles of high dynamic range using differently exposed pictures were inspired by the pioneering work of Charles Wyckoff [38] with the invention of *"extended response film"*, which consisted of three layers of different light sensitive films. The final image is a composition of these layers. Figure 1.3 (b) shows an example where this technique was used to photograph a nuclear explosion. To translate this design into digital cameras, we need to temporally capture multiple exposures of the scene where the final image is created by fusing these images using some weighting filter based on the noise characteristics or nonlinearities in the captured images. Despite the prevalence of this method, it is mostly suitable for static environments, as any movements in the scene will result in a blurry image that requires tedious image registration techniques [39]. Therefore, in most cases, these methods fail to capture real scenes with moving objects accurately and are not suitable for dynamic environments.

One of the main challenges in HDR imaging is to acquire the HDR image in a single capture setting to avoid the ghosting artifacts introduced by temporal multi-exposure techniques [40, 41]. An HDR image can be captured in a single shot by combining a multi-sensor setup with a beam splitter and *Neural Density* (ND) filters to change the exposure on each sensor[19]. Nevertheless, the required types of equipment for this method are usually costly and not suitable for hand-held consumer cameras.

Single HDR image acquisition is also possible by spatially multiplexing the exposure by placing an ND-filter in front of the sensor or lens. However, ND-filters can cause color shifts, and since they limit the effective light throughput, they lead to image noise in the final result. Another way is to utilize multiple gain settings in each capture where gain values amplify the analog signal before passing to the A/D converter unit. However, in this way, the noise is also amplified, which results in a grainy image. There are a variety of algorithms proposed for reconstructing an HDR image from spatially interlaced exposure input, which mainly focuses on recovering each exposure setting separately and using some weighting function to fuse them to achieve an HDR image. However, these techniques will result in low spatial resolution in the final reconstructed image since they do not employ the information from other exposure settings in the reconstruction process. In this thesis, the focus is on the capture and reconstruction of HDR images using spatially multiplexing by varying (dual) ISO settings through a single image capture. Our proposed method recovers a full-resolution HDR image in a single-step procedure from a dual-ISO image. More details are explained in Chapter 3.

1.1.2 Light Field Imaging

Light field imaging is a powerful technique striving to capture the spatial, as well as the angular, radiance information of a scene, while enabling sophisticated digital processing methods for this high dimensional data. Light field imaging has a history of more than 100 years with the first lenticular parallax stereogram introduced by Fredric Ives [42] in 1903, followed by the first practical design for a light field camera from the Nobel prize-winner Gabriel Lippman [43]. Lippman proposed a model to place a lenticular multi-array lens in front of the primary lens to capture light rays from different directions. The term "*light field*" was first coined by Gershun [44] in 1936. The computer vision community was introduced to the concept in 1992 by Adelson and Bergen [45] with the plenoptic camera design, which was later on implemented and improved by Ng et al. [46] as a hand-held plenoptic camera. Only in the past decade, there have been significant advancements in light field imaging concerning capturing, compression, and rendering.

Multiple designs have been suggested to sample the incoming illumination, including multi-camera systems [47], microlens arrays placed between the primary lens and the sensor [46], and gantry systems [48]. Each design is suitable for a specific application and scene characteristics (e.g. static or dynamic). For instance, multiple camera systems are usually suitable for dynamic scenes, while the gantry system can capture high angular resolution of a static scene. Figure 1.4 shows a summary of advancements in light field imaging. An overview of the field can be found in Chapter 5.

Figure 1.4: *The timeline of the development of light field imaging.*

One of the main challenges in light field imaging is that capturing high angular resolution light fields requires a lot of data, meaning that the capturing, processing and storage of the data becomes very challenging. New light field camera designs can reduce the required number of samples in the capturing phase using e.g. *compressive sensing* to alleviate the excessive bandwidth requirement of light fields. Additionally, effective compression techniques can encode these data even further in a way that they can be transferred over a network or fit on the GPU for real-time rendering. This thesis proposes a set of novel techniques to obtain a framework for capturing, compression, and rendering of 4D-6D light field data sets with significant improvements over the state-of-the-art methods.

1.1.3 Applications

Digital images assist humans and machines in a variety of applications. Image-based rendering can be considered as the first application of digital imaging. IBR techniques are a compelling alternative to geometry-based approaches in image synthesis. Figure 1.5 shows a continuum of the image-based representations that depend on the available geometric information versus the number of input images. Pure image-based techniques rely only on the photos or videos captured from the scene. However, depending on the environment and its complexity, additional information is sometimes required for seamless renderings, such as implicit geometry information like depth or disparity map. In the rightmost end of this continuum is rendering with available explicit geometry and scene details like material properties, and illumination information. This type of image synthesis is known as *physically based rendering*, an accurate but computationally expensive technique.

HDR rendering or lighting is an IBR technique where the lighting information is stored in an HDR domain. Utilizing this lighting domain in rendering allows us to preserve the details of the scene that might be lost due to limiting contrast ratios. HDR images of the environment illumination, often called *radiance maps*, are used to illuminate virtual 3D objects, a practice with many advantages in the

Figure 1.5: *Image based rendering (IBR) versus Physically based rendering(PBR). The left of the spectrum (IBR) uses less geometry information and the right side of the spectrum uses more complex geometry and simpler image forms.*

photography and film industry [28], as well as in image-based lighting [49] and material editing [50]. Creating a virtual world using an HDR rendering is a more realistic and compelling solution compared to modeled lighting due to the accurate simulation of the physics of light. Figure 1.6 (a) shows an HDR video camera and light probe, where the HDR camera can capture up to 24 f-stops of the scene. The HDR environment map sequence from this camera is employed to illuminate a virtual object placed in a real environment, as seen in Figure 1.6 (b).

HDR imaging has been used for photography of scenes with complex illumination as well as in computer vision and image processing techniques to better find features in the captured images. Moreover, with the development of HDR displays, HDR content is necessary to bring entertainment experiences closer to reality. HDR imaging has also been employed in the automotive industry for advertisement, where a new car model is rendered with real-scene illumination before massive production. The vehicle can also be equipped with an HDR camera to improve the safety of autonomous driving in all lighting conditions.

The recent advancements of light field imaging have affected various disciplines such as computer vision, computer graphics, medicine, and biology, and demonstrated more promising results when compared to traditional imaging techniques. Light field imaging in computer graphics is a crucial tool for rendering, as it is a process independent of the geometry, and complex material representations [51, 52]. From a different perspective, one can employ geometrical information such as depth map or optical flow to reconstruct novel views and extend the angular resolution of a light field. Refocusing an image is yet another application of light field imaging where the depth of field is changed in a post-capturing process [21].

The light field embeds a significant amount of information (e.g. depth), which can be explored as an *epipolar plane image* (EPI) to increase the angular resolution by generating novel views [53]. Additionally, in the field of computer vision, light field imaging can be employed to estimate scene flow and material identification more

Figure 1.6: (a) HDR Video camera developed in collaboration between SpheronVR and Linköping University that enables panoramic lighting measurements [9]. (b) Rendering virtual objects illuminated with the footage from the HDR video camera [7].

robustly, and regardless of illumination variations [54]. The material properties can also be approximated in terms of spatially varying bidirectional reflectance distribution function (SVBRDF) using light field images [55]. In addition, many other fields can benefit from light field imaging techniques. For instance, microscopy [56], vision-based robot control [57], and bio-medicine tools such as otoscope [58].

1.2 Objectives and Contributions

The principal research objective of this thesis is to advance the capabilities of ordinary cameras to be able to capture more enhanced information about the scene. This includes, but is not limited to, higher dynamic range and higher dimensional data such as light fields. Advancing the state-of-the-art in these areas requires solving the following problems:

- A statistical noise model suitable for reconstructing an HDR image from a spatially multiplexed image with dual gain settings.
- A framework for jointly solving multiple problems such as denoising, demosaicing, and HDR fusion in a single step for spatially multiplexed images.
- Designing a framework for encoding and real-time reconstruction and rendering of light fields and light field videos that is robust to the noise.
- A design for capturing light field video in the compressive setting.

The research carried out by the author of this thesis has led to the following contributions:

Increasing the dynamic range of consumer cameras. A unified kernel regression algorithm based on the statistical noise model of the camera is introduced for the reconstruction of interlaced multiexposure images (*Paper A*) where the processes of denoising, reconstruction, and demosaicing are performed jointly in a unified manner. The adaptive kernel preserves the structure of the image while it instantaneously removes noise according to image statistics and camera noise model (*Paper B*).

Compression and rendering of light field images and videos. A robust framework is designed to compress high resolution light field data sets such that the encoded coefficients can fit easily on the GPU memory. The framework is based on learning a multidimensional dictionary ensemble (MDE) to exploit the sparsity of the natural light field data sets. To improve the computational complexity of the training stage, and the sparsity, a novel pre-clustering algorithm based on an ℓ_0 pseudo norm is applied to the training set. The obtained MDEs from each pre-cluster, are collected into an aggregated multidimensional dictionary ensemble (AMDE) that is used to compress the data (*Paper C*). To improve the compression performance, the AMDE is pruned with a novel algorithm that finds the most distinct dictionaries in the AMDE (*Paper D*). A real-time reconstruction algorithm is introduced in *Paper C*, where each element of the light field data points are reconstructed independently. The real-time rendering method is extended in *Paper D* with novel view synthesis. The effect of camera noise on the compression algorithm is extensively studied and demonstrated that even with slight denoising before encoding, the compression ratio improves significantly. Finally, in *Paper E*, a new design for capturing spherical light field is introduced, and we show that the AMDE framework is suitable for compression and real-time rendering of such massive amounts of data. The quantization and entropy coding of sparse coefficients obtained from the AMDE framework have also been studied for spherical light field data.

Compressive light field video camera. A novel reconstruction algorithm is presented to recover the light field video data captured with a coded-aperture design (*Paper F*). A color-coded mask is placed between the sensor and the aperture plane that convolves the incoming light rays into a single 2D image. The high-resolution light field video is reconstructed by employing a novel compressive sensing model using a dictionary-learning algorithm that considers spatial, angular, spectral, and temporal domains.

1.3 Thesis Outline

The rest of this thesis is structured as follows: Chapter 2 provides a brief and necessary background, as well as technical information, about the fundamentals of the camera and its properties. Chapter 3 presents details of the camera noise model and the HDR reconstruction algorithm for spatially multiplexed data. The sparse representation of the signal is explained in Chapter 4 that covers the basic concepts of dictionary learning and compressed sensing. Chapter 5 details our framework, known as AMDE, for the compression and reconstruction of light fields and light field videos. The main ideas of the compressive light field video camera are explained in Chapter 6. Finally, Chapter 7 provides a conclusion and a summary of the topics presented in the thesis, as well as a discussion of future directions for the research.

Fundamental Camera Concepts

Photography is about *seeing* the world through a camera lens, by capturing the light arriving at the camera sensor. The photographer controls the amount of light reaching the sensor by tuning a set of parameters such as exposure time, aperture, and ISO. Understanding the fundamental concepts of digital cameras is necessary in order to develop computational methods that enhance the capturing process. This chapter describes the basic computational photography concepts that have been used throughout this thesis. These include an explanation of the digital camera pipeline, various camera parameters, and noise characteristics in the capturing process. Furthermore, the dynamic range of the human visual system and digital cameras are compared.

2.1 In-Camera Processing

There are primarily two technologies used for the camera sensor to acquire a digital image: charge-coupled devices (CCD) and complementary metal-oxide-semiconductors (CMOS). Although the operational principles of these two sensor designs are different, their acquisition models are similar in the sense that both are converting the incoming light photons into electric voltage values. In this section, the acquisition process is explained in details, including the camera parameter description, as well as various sources of uncertainty that can be introduced at each stage of the acquisition.

Figure 2.1: *RGB color filter array (CFA) on a digital sensor camera with RGGB pattern. Each element of the CFA passes through the light rays in the same spectrum as the filter resulting in an RGB image.*

2.1.1 Color Filter Array

In conventional cameras, a color filter mosaic, generally called *Color Filter Array* (CFA), is placed on top of the imaging sensor, resulting in a mosaiced image. The most common CFA is with RGB colors, which is called Bayer pattern [59]. Color filter arrays such as the Bayer pattern are based on the sensitivity of human vision to the green wavelength rather than other colors, and it is designed in a quad of pixels that follows a pattern of 25% red, 25% blue and 50% green. Each photosite only samples one color. Figure 2.1 illustrates a CFA with an RGGB pattern placed in front of the sensor where only photons with the same wavelength as CFA pass through and are converted to electrons. The acquired image is spatially under-sampled in color channels, and requires reconstruction through an interpolation process called *demosaicing* or *debayering* [17].

Other types of CFAs also exist, such as CMY (Cyan, Magenta, Yellow) CFA, RGBE (Red, Green, Blue, Emerald) CFA, and CYGM (Cyan, Yellow, Green, Magenta) CFA. Different designs for the CFA pattern have been proposed, such as RGBW (Red, Green, Blue, White) CFA [60], a perceptually based design with a random pattern [61], designs robust to noise [62] and many more. Throughout this thesis, the assumption is that the camera is equipped with a CFA with a Bayer pattern.

Demosaicing algorithms aim to recover the full-color image from the spatially incomplete color samples of the image sensor overlaid with a CFA. Demosaicing algorithms can be categorized as following: interpolation-based methods and learning-based methods. Simple interpolation of color channels results in color artifacts, especially around the high-frequency edges. Considering the local features of the image and utilizing the correlations among color channels leads to better results [63, 64, 65, 66]. However, these methods fail to recover the signal without artifacts in complicated structures.

Learning-based methods, on the other hand, learn the local distribution of the image patches and provide better color fidelity in the reconstructed image [67, 68]. Deep convolutional neural networks (CNNs) have recently shown promising results for joint denoising and demosaicing of the sensor output, where the visual artifacts are reduced compared to traditional methods [69]. For a comprehensive overview

of these methods see [70].

2.1.2 Camera Parameters

There are a number of parameters on the camera that constrain the way an image is captured. These parameters control the brightness of the image, the depth of focus, sharpness, and noise. A correct combination of them allows capturing a properly exposed image. However, each of these elements can also introduce some side effects that should be taken into account during the imaging process. This section explains a few of the essential parameters that are used in the computational algorithms.

Shutter speed Shutter speed or *exposure time* is the length of time that the sensor is exposed to incoming light. The amount of light arriving at the sensor is proportional to the exposure time, meaning that more photons reach the sensor by increasing the exposure time.

Aperture The optical system of the digital cameras consists of lens components and an aperture that controls the amount of light that arrives at the imaging sensor. The aperture and the focal length determine the bundle of rays that are in focus on the image plane. The aperture controls the brightness and the depth of field of the captured image. A large depth of field means that most of the image is in focus while the shallow depth of field means that only a small part of the image is in focus. Opening the aperture increases the brightness of the image while creating a shallow depth of field.

ISO/Gain *Gain* is proportional to the *ISO sensitivity* on modern cameras, which allows the signal to be amplified before analog-to-digital (A/D) conversion. ISO is comparable to film sensitivity in analog cameras. Increasing ISO will amplify both the signal and the noise in the signal, which can result in a grainy image.

Exposure There are different ways to adjust the exposure: one method relies on modifying the shutter speed or the integration time. Short integration times capture bright parts in the scene well, while longer integration times are suited for capturing dark regions in the scene. The latter can lead to motion blur. Another approach is to change the aperture, which affects the image focal length, a not always desirable effect. ND-filters can be used to change the exposure in all or parts of the image. Even though they are effective in capturing dynamic scenes, blocking the incoming luminance can increase the noise in the dark areas of the scene. Another approach is to change the gain or ISO setting, which can have the benefits of avoiding motion blur and improving the focal length.

Figure 2.2: Sources of noise in the capturing process of the radiant power from the scene to digitized signal values. The figure is inspired by [71].

2.1.3 Camera Noise Sources

When a sensor measures light, there are several sources of noise that can corrupt the signal in different stages of capturing. Understanding these sources can help in estimating the underlying signal more accurately. Figure 2.2 shows different sources of noise that appear in different stages of digital imaging. The sensor noise can be stationary fixed-pattern noise, or vary over time. Fixed-pattern noise occurs due to irregularities during the manufacturing of the sensor that introduces unwanted spatially varying noise and remains the same regardless of the signal or the capturing time.

Photo-response non-uniformity (PRNU) Theoretically, each pixel should collect precisely the same amount of photons when uniform light falls on the sensor. However, due to variations in the substrate material, and pixel geometry (e.g. pixel size), the output values are slightly varied over the sensor pixels. The difference between the expected response from uniform light and the actual output value from the sensor is called PRNU. Removing PRNU is almost impossible since it is caused by the physical properties of the sensor. However, the effects of this noise can be reduced by using a lookup table (LUT) made of correction factors for each pixel that is calculated by exposing the camera sensor to a uniform light source.

Dark signal non-uniformity (DSNU) This type of fixed-pattern noise occurs because of variations in the sensitivity of each pixel for collecting photons due to fabrication errors. This noise can be estimated by averaging multiple photographs taken with fixed lens cap. Some cameras have built-in functionality to remove this noise by taking a second exposure with the shutter closed and subtracting it from the real captured image.

Photon shot noise The number of photons reaching the sensor changes for every capturing, and it results in fluctuations in the captured signal known as photon shot noise. The time of arrival of each photon is random, and it follows a

Poisson distribution indicating that if the number of photons increases, the variance in the signal also increases by the same amount [72].

Dark current noise This noise is introduced due to the thermal electron vibrations on the sensor, especially in low-light photography, such as in astronomy, when a longer exposure time is required. The effect of it can be reduced by cooling down the sensor, and in most applications, it can be neglected for exposure times less than one second¹. Dark current noise or *dark shot noise* is independent of the number of photoelectrons generated, which means it is an additive noise and a Poisson distribution can model it well [73].

Reset noise Before capturing an image, the photosite wells need to reset from any charge or electrons. However, if the required time for resetting each well is longer than the clock speed of the camera, some wells might not be reset when the next capturing begins, meaning that some wells might contain electrons from previous capturing. This residual charge creates a spatially varying signal, which is known as reset noise [74]. This noise, similarly to DSNU, can be accounted for by capturing a bias frame where after each pixel is reset, the remaining signal is read. The noise is removed by subtracting the bias frame from each captured image. The bias frame is acquired by taking an image with the cap on, resulting in the offset values for the pixels of the image. The reset noise can be accurately modeled as a Gaussian distribution [75].

Readout noise During the readout stage of the camera pipeline, the electron charges gathered in each photosite are read. The readout noise is created between the photoreceptors and the A/D converter circuitry and is thermally generated, which is modeled as Gaussian noise [76]. In the CMOS sensors, the readout phase is done by reading the pixels line by line, which introduces a patterned noise that can be disturbing to the human eye [76].

ADC noise In the quantization process of the A/D converter, where the analog voltage measures are converted to digital values, a uniformly distributed quantization error is introduced in an additive fashion. Compared to other sources of noise, ADC noise can be neglected [77].

Removing the noise introduced during the capturing process requires solving an inverse problem with an infinite number of solutions. Since noise, edges, and textures are high-frequency features, it is difficult to distinguish between them for removing the noise. As mentioned above, some sources of noise are signal-dependent, i.e. multiplicative noise, while other sources are signal-independent or additive.

¹ <http://theory.uchicago.edu/~ejm/pix/20d/tests/noise>

Figure 2.3: Real world luminance and sensitivity of the human visual system (HVS), digital cameras and HDR exposure bracketing.

The major challenges of image denoising algorithms are to keep the homogeneous regions as smooth as possible, while the structure of the edges remains intact, and no new artifacts and blurs are introduced.

2.2 Dynamic Range

The human visual system (HVS) is capable of perceiving illumination in a range of approximately 10^{14} , from 10^{-6} to 10^8 cd/m^2 by transforming the incident light on the eye into nerve impulses using photo-receptors. There are two categories of photo-receptors in the human visual system: *Rods*, which are more sensitive to the light, enabling vision in dark environments with illumination ranging from 10^{-6} to 10 cd/m^2 , e.g. night scenes. This is called *scotopic* range when only rods are active. *Cones* are another category of photo-receptors that are less sensitive to the light, and they are active during daylight conditions, forming the *photopic* vision. Rods have multiple receptors that are sensitive to long, medium, and short wavelengths, while cones can only register monochrome information. The rods are effectively saturated in photopic vision, ranging from about 0.03 to 10^8 cd/m^2 . The range that both rods and cones are active is referred to as *mesopic* vision which extends from about 0.03 to 3 cd/m^2 [28]. Figure 2.3 depicts the sensitivity of the human visual system with scotopic, mesopic, and photopic vision at different light levels. Photopic vision corresponds to normal light levels where rods saturate, and cones dominate. The scotopic vision resembles low light levels where rods dominate due to the lack of sensitivity of the cones.

This extensive dynamic range of illumination allows humans to see objects under sunlight as well moonlight. However, the HVS cannot operate over such an extensive range simultaneously. This significant variation in sensing luminance is accomplished by brightness adaptation. Therefore the simultaneous range that humans can sense is rather small and in the range of 10^3 cd/m^2 at any particular state of adaptation.

On the digital cameras, the dynamic range is dependent on the bit-depth of the sensor, which means that for a camera with 14 bits per pixel, it can measure up to 2^{14} photons. However, in reality, this range is lower, and the dynamic range is dependent on the size of each photosite or pixel capacitor and its sensitivity to light. Each photosite transforms photons that hit the corresponding pixel on the sensor to electrons (charge), and it overflows and saturates if its well capacity is reached. Furthermore, the sensitivity of the photosite further determines the darkest measurable light intensity. Therefore the dynamic range of a digital camera is usually calculated as a ratio between the highest number of photons that each photosite can contain and the darkest measurable light intensity. The most common unit for measuring the dynamic range of the camera is the f-stop. For example, a contrast ratio of 1024:1 is translated into 10 f-stops (2^{10}).

2.3 Quality Metrics

The quality of a reconstruction algorithm for image processing tasks such as denoising, demosaicing, needs to be evaluated by a well-established quality metric. Although the best way to assess the quality of an image is by a combination of objective and subjective quality metrics, this thesis considers only the objective metrics for evaluation as they provide accurate measures for the stated problems.

2.3.1 PSNR

The objective measure that is most commonly used in image and video processing is *Peak-Signal-to-Noise Ratio* (PSNR), which is defined for an image I_r and its approximation I_t as follows:

$$PSNR(dB) = 10 \log_{10} \frac{L_{max}^2}{MSE}, \quad (2.1)$$

where, L_{max} is the maximum possible pixel value of the image and MSE is the mean square error calculated as follows:

$$MSE = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n (I_r(i, j) - I_t(i, j))^2. \quad (2.2)$$

2.3.2 SSIM

Another metric is the structural similarity index (SSIM) [78] which is a method for measuring the similarity between two images. The quality of this metric is reportedly corresponds better to perceived differences than MSE and PSNR [79]. The SSIM is calculated between two windows of size $N \times N$ pixels within the original signal $\mathbf{x}$ and the reconstructed signal $\mathbf{y}$ as follows:

$$SSIM = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \quad (2.3)$$

where μ_x and μ_y are the mean of $\mathbf{x}$ and $\mathbf{y}$, respectively, and σ_x^2 and σ_y^2 denote their corresponding variances. The covariance between two matrices is denoted as σ_{xy} . The two variables c_1 and c_2 are constants that prevent the numerical instability and are defined as:

$$\begin{aligned} c1 &= (k_1 L)^2 \\ c2 &= (k_2 L)^2, \end{aligned} \quad (2.4)$$

where L is the dynamic range of the image and $k_1 = 0.01$ and $k_2 = 0.03$.

2.3.3 HDR Quality Metrics

High dynamic range imaging algorithms can be evaluated objectively, or subjectively. A human observer can detect the visual differences between the results of two algorithms and choose the one that most resembles the reference image. However, subjective evaluation is not always practical and requires an extensive amount of experiments. Therefore, it is necessary to find a quality metric that is suitable for these computational tasks.

High dynamic range pixel values correspond proportionally to the physical luminance. In contrast, the low dynamic range pixel values are usually gamma-corrected and relate nonlinearly to the luminance of the scene. The gamma curve approximates the response of the human eye to luminance. The HVS is more sensitive to the luminance ratios than absolute values (Weber-Fechner law). Consequently, the direct comparison of HDR pixel values does not represent what is perceived by the human eye. Quality metrics for HDR algorithms should compensate for this effect to provide a reliable evaluation. One approach is to transform the HDR values into a domain representing an HDR or LDR display, such as perceptually uniform algorithms [80]. Other methods convert HDR pixel values into a domain that mimics the human perception of luminance [81].

A common method is to use nonlinear luminance metrics where the HDR pixel values are encoded to a domain linear to the HVS, such as the logarithmic domain.

Figure 2.4: *Perceptually uniform (PU) quality metric.*

The LDR quality metrics such as PSNR or SSIM can then be applied to the logarithmic values [82]:

$$\log PSNR = 10 \log_{10} \frac{\log_{10}(L_{max})^2}{MSE} \quad (2.5)$$

$$MSE = \frac{1}{N} \sum_{i=1}^N [\log_{10}(\hat{I}_t(i)) - \log_{10}(\hat{I}_r(i))]^2, \quad (2.6)$$

where,

$$\hat{I}_t(i) = \max(I_t(i), L_{min}), \quad \hat{I}_r(i) = \max(I_r(i), L_{min}), \quad (2.7)$$

where L_{max} and L_{min} are the peak luminance level, and minimum luminance above the noise level, respectively. The distorted image is denoted as I_t and the reference image as I_r . Similarly, SSIM can be estimated in the logarithmic domain. One can also normalize the log-transformed HDR pixel values before calculating the SSIM or PSNR[83].

The HDR pixel values can also be encoded by applying a transformation into a domain where encoded pixel values correspond to a perceptually uniform (PU) representation [80]. The encoding is derived from the contrast sensitivity function (CSF), which describes the capacity of the HVS to predict luminance and chrominance differences as a function of contrast and spatial frequency. The transformation is further constrained to imitate sRGB nonlinearity. The encoded values are then passed to an LDR metric such as SSIM, see Figure 2.4. The resulting quality metric is known as *PU-SSIM*.

A more comprehensive encoding model of the contrast visibility that takes into account luminance dependent effects like intraocular light scattering is HDR-VDP-2 [81]. This quality metric is also derived from a measured CSF and predicts the visual differences at all range of luminances. HDR-VDP-2 calculates the probability of detecting a difference between two HDR images, as well as the perceived level of distortion.

High Dynamic Range Imaging

In a two-dimensional imaging system, the photons are integrated from multiple angles during a specified time (exposure) and absorbed by the camera sensor's capacitors. The limitations of the capacitors indicate the dynamic range of the sensor. If the number of photons exceeds the limitation, it overflows (the pixel saturates), and the information in that region becomes unreliable. The dynamic range is defined as the ratio of the maximum and the minimum number of photons a sensor can gather at each pixel. A scene's dynamic range is represented by its contrast ratio from the darkest shadows to the very well lit areas. A typical scene may exhibit a linear dynamic range in the order of 10,000,000:1. In contrast, the maximum dynamic range a camera can measure nowadays is up to 16.5 f-stops, which corresponds to a contrast ratio of around 92,000:1, for cameras such as RED Helium 8K and ARRI Alexa¹. Depending on the ISO sensitivity of the sensor, clamping, and noise, the useful range is far less in reality.

Consequently, even the high-end cameras are incapable of capturing the full dynamic range and radiance distribution of a scene. Unfortunately, with the current development of the sensors, it is not possible to obtain the human dynamic range in a single shot. As a result, there is a demand for computational methods together with new system designs to overcome these limitations. This chapter presents an overview of the computational techniques for capturing and reconstructing HDR images. Furthermore, it describes details of the statistical algorithms for single-shot spatially interlaced HDR image reconstruction, as presented in *Paper A* and *Paper B*.

¹ www.dxomark.com

3.1 Outline and Contributions

This chapter presents two algorithms for extending the dynamic range of conventional cameras in a unified framework for reconstruction, demosaicing, and denoising based on the statistical information acquired from the camera noise model. The main contributions of the research presented in this section are related to the statistical reconstruction [9] of interlaced data with multiple gain settings, using Local Polynomial Approximation (LPA) [84]. As the first contribution, *Paper A*, a filter kernel based on LPA is applied to a spatially interlaced image with two different gain values, where the dynamic range of the reconstructed image is extended by about 2-3 f-stops. In the second contribution, *Paper B* extends this approach by adapting the filter kernel to the local image statistics, the sensor noise, as well as the underlying signal structure present in the image. The adaptive filter kernel increases the image quality in various examples presented here.

This chapter is organized as follows: Section 3.2 provides an overview of the acquisition methods, where three categories of techniques are discussed. Section 3.3 explains various HDR reconstruction approaches, which is followed by Section 3.4, where the details of the two statistical HDR reconstruction frameworks are explained. Finally, Section 3.5 concludes the chapter by a summary of the presented methods and the possible future venues for research.

3.2 HDR Acquisition Overview

High dynamic range imaging (HDRI) has been a very active and well-studied research topic in computational photography. HDR image acquisition has many applications from digital photography, image-based lighting [49], digital cinema, computer games, virtual reality, and HDR display [85]. For an extensive overview of HDR algorithms for capturing and reconstruction see [28]. Various approaches for capturing HDR images can be broken into the following categories: the first category is multiple exposure acquisition using a single sensor, the second one is single capturing with multiple sensors, and finally single shot with a single sensor. Acquiring HDR videos using conventional cameras and mobile phones is still an ongoing research problem. This section presents an overview of the state-of-the-art methods for capturing the HDR content of natural scenes.

3.2.1 Multi Shot - Single Sensor

Temporal multi-exposure capturing, known as *Exposure Bracketing*, is a very well-researched and well-established method that is available on most consumer cameras. The idea was reintroduced to the digital photography field by Madden [86] in 1993. In this approach, multiple low dynamic range (LDR) photographs

Figure 3.1: *Exposure bracketing for HDR capturing. The highlights are captured with low exposure while the dark regions of the scene are captured with high exposure image.*

are sequentially captured with different exposure times. These LDR images are fused into a single, high contrast HDR image as a weighted average of the pixel values across the different exposures, by taking the camera response function into account [40, 41, 87, 88]. A very wide dynamic range can be captured with this approach since one can take many images of the scene, where each image is exposed such that a different luminance range is acquired. This technique is now available on most consumer products such as smartphone cameras and is known as *HDR mode*. The *HDR mode* usually works by blending several images taken at different exposures, and its blending weights are based on measures of quality such as color distribution or local contrast [89]. An example of exposure bracketing is shown in Figure 3.1, where three exposures are combined to create a full dynamic range of the scene. The short exposures, i.e. the top right image, captures the highlights of the sky while the long exposure image, i.e. the bottom right, is capable of recoding darker areas of the scene.

Despite its popularity, exposure bracketing is not capable of capturing dynamic scenes and motions. Any movements in the scene will result in blurring and ghosting artifacts, which requires advanced methods to recover a sharp image. Several approaches have been proposed to overcome these issues by aligning the images from a stack of multiple exposures to a reference image and eliminating any misalignments before reconstruction. These methods are usually dependent on correspondences at pixel-level [90, 91, 92] or patch level [93] and optical flow [94], while others take into account per-pixel values as a blending parameter

Figure 3.2: *Left: A multisensor HDR video camera developed by Spheron VR and Linköping University, with ND-filters and beam-splitters. Right: A light efficient multisensor design proposed by Tocci et al. [19].*

[28]. Nevertheless, finding correspondences between the images is computationally expensive and an error-prone task, especially when there is a large shift between the corresponding points.

Deghosting the HDR images using rank minimization [95, 96] is another approach for removing the artifacts from the reconstructed image. A patch-based method with energy-minimization can potentially align multiple exposures and reconstruct the HDR image using a joint optimization model. Despite all the advances, these methods often fail to robustly handle movements in a general scene due to parallax, occlusion, and deformable motions. For an overview of these methods see [97].

Recently, deep learning methods have tried to overcome the dependency on the correspondences for alignment by hallucinating the details of an HDR image in the absence of information caused by e.g. occlusion, saturation, and under-exposed regions [98]. Nonetheless, these methods are unable to remove the ghosting artifacts in the final reconstruction altogether.

3.2.2 Single Shot - Multiple Sensors

Another method for capturing HDR content is using beam-splitter and multiple sensors, which has shown to be useful for dynamic scenes and also capturing HDR video [19]. Aggarwal et al. [99] suggested a design where the aperture is split into multiple parts, and the incoming ray is redirected into different beams. Each beam moves towards a sensor where the exposure is either controlled by setting parameters or by splitting the incoming ray unevenly.

Spheron VR and Linköping University employed a similar method to develop an HDR video camera that is capable of capturing 24 f-stops at 30 frames per second [9], see Figure 3.2 for the prototype design. To control the amount of light reaching each sensor, they utilized ND-filters with different sensitivity for each sensor. The

significant advantage of these systems over temporal exposure fusion HDR methods is that the motion in the scene is handled very robustly as all the sensors have the same exposure time. However, using these systems, the light efficiency decreases due to ND-filters blocking light rays, which introduces noise in dark regions of the image. Tocci et al. [19] suggested an optimized design using a beam splitter setup that increases the light efficiency, as shown in the right image of Figure 3.2.

Nevertheless, another disadvantage of these methods is that they require high precision in the optical design and accurate geometrical alignment of different sensors [9, 99, 100]. Furthermore, the hardware and the prototype setup are generally expensive and challenging to make available to consumers. The dynamic range captured by these systems is limited to the number of sensors that can be used. Any additional sensor complicates the system’s optical light splitting design and reduces the adequate light reaching each sensor.

3.2.3 Single Shot - Single Sensor

The final acquisition category relies on using only a single sensor, where spatial multiplexing is employed to capture an HDR image. The research presented in this thesis falls into this category of techniques. There are two main factors to consider for designing a spatially varying system. Firstly, the dynamic range is dependent on the number of different exposures that are captured in a single shot. A higher number of spatially varying exposures increases the dynamic range while reducing the spatial resolution of the reconstructed image. Secondly, the spatial distribution of these varying per-pixel exposures, i.e. regular or random, affects the reconstruction differently. Regular patterns introduce an aliasing effect, especially around high-frequency features of the image, e.g. edges. Although a random pattern can alleviate these artifacts, they are harder to implement in practice. One way to vary the exposure in a single image is to place a spatially varying ND-filter in front of the sensor with regular [18, 101] or irregular patterns [102, 103]. However, these methods lead to low signal-to-noise ratio due to the light inefficiency of the ND-filters.

A theoretical design called *Gradient Camera* was introduced in 2005 by Tumblin et al. [104], which is based on capturing the gradients of the image rather than the actual pixel intensity and later applying an expensive Poisson solver to reconstruct the intensity values from it. In *Programmable Imaging* [105, 106], an aligned spatial light modulator, such as Digital Micromirror Device (DMD), was used instead of ND-filters and provided adaptive control over the exposure of each pixel. This method’s main disadvantage is the pixel-level alignment of the DMD with the sensor, which is a laborious task [107]. Nayar and Branzoi [108] proposed an alternative implementation of the spatial exposure variation, called *Adaptive Dynamic Range Imaging* (ADRI). They mounted a transmissive spatial light modulator near the

aperture plane of the camera to simulate an adaptive optical density mask instead of a fixed-pattern element. However, this method only modulates the low spatial frequencies in the image.

The coded rolling shutter photography [48, 109, 110] controls the readout timing and the exposure time for each row of the CMOS sensors to enhance the captured image by deblurring, skew correction and extending the dynamic range. Reconstruction of HDR image requires the calculation of optical flow, cubic interpolation of each exposure time, and fusing them, which is computationally costly. Another method is to use a spatial light modulator to create a random mask pattern in front of the sensor to encode the light rays arriving at the sensor. Afterward, a sparse convolutional coding method can be applied to the modulated image to reconstruct the HDR image [111].

More recently, with the advancements in the CNNs, algorithms have been proposed that map an LDR image to its corresponding HDR one by recovering information from the saturated regions [112] and extending the dynamic range. Similarly, CNNs can be used to train a single image contrast enhancer where the training set is created from multiple exposure images [113]. These algorithms are mainly focused on recovering saturated information, which limits their capabilities.

An alternative solution is to use all incident light using imaging sensors where the pixel gain is varied over the sensor [48, 109, 114], such that the low gain setting captures the bright regions of the image, by avoiding saturation, while the high gain setting has a better signal-to-noise ratio in the dark areas in the image. This method can be applied to existing off-the-shelf digital cameras equipped with a customized firmware update (e.g., Magic Lantern firmware for Canon cameras). These camera sensors are then capable of using two ISO settings simultaneously by varying the gain for every other row, covering the RGGB Bayer pattern for each exposure setting, as depicted in Figure 3.3. Some sensors such as Aptina AR1331CP, Sony IMX135, and Apertus Axiom are also capable of recording two exposure settings for every other row (every two rows of the raw Bayer image covers all the color samples). This chapter introduces methods for reconstructing an HDR image using the output of such capturing systems.

3.3 HDR Reconstruction Overview

Early HDR reconstruction techniques focused on fusing differently exposed LDR images using the response curve of the sensor acquired from a calibration process [41, 87, 88]. Later on, some methods consider weighting the pixels by exploiting the noise characteristics in the linear domain to minimize the variance of estimation [115]. Granados et al. [116] takes into account different sources of sensor noise,

Figure 3.3: Gain patterns for Dual-ISO setting. The left image shows a unified reconstruction kernel applied on the green color channel for this gain pattern.

i.e. temporal (photon, dark current shot noise, readout noise) and spatial (Photo-response and dark current non-uniformity) as a function of irradiance reaching the sensor. Gallo et al. [117] analyze the image histogram, and adaptively select a minimal number of exposures to capture the scene with an optimal signal-to-noise ratio. By taking into account different sources of sensor noise, Hasinoff et al. [71] introduced a method to minimize the variance of the estimation.

The emphasis of most HDR acquisition methods is on fixing the ISO/gain settings to a low value to reduce the noise. However, it has been shown that sometimes high ISO is desirable and can lead to a better signal-to-noise ratio of the final image [118, 119]. Higher gain value and lower exposure time have been used to eliminate and reduce the ghosting artifacts of the time-sequential multi-exposure methods [120].

Reconstructing high dynamic range images and videos demands solving various tasks such as reconstruction of full color from CFA data, alignment of the images for multi-sensor algorithms, and HDR assembly, where low dynamic range (LDR) images are fused to produce the HDR content, and finally a denoising step to reduce the excessive noise introduced in the capturing process.

Traditionally, HDR reconstruction algorithms [121] adopt a pipeline where each of these tasks are solved independently from others, as shown in the top image of Figure 3.4. Splitting the processing unit into individual tasks poses several issues and shortcomings. Each process requires prior assumptions for its solution and introduces an error which is accumulated throughout the pipeline. More importantly, all of these problems are correlated, and there is no natural order in solving them.

As an example, some reconstruction methods propose to solve demosaicing before HDR assembly [41], which causes artifacts, especially in saturated regions where there are not enough samples. Furthermore, some color channels saturate at different

Figure 3.4: Traditionally HDR reconstruction methods proposed a reconstruction pipeline consisting of independent processes such as demosaicing, resampling, HDR assembly, and denoising. A unified algorithm combines all these steps into one simultaneous task.

levels, depending on the scene and its illumination. Performing demosaicing after HDR assembly [19, 122] creates problems such as blurring, especially if the input data is captured with exposure bracketing or multi-sensor devices.

Solving multiple tasks in a joint approach has shown to be more accurate and efficient when compared to the traditional pipelines. An example is joint demosaicing and denoising [123, 124], which results in a sharper image, especially in low light conditions. Other examples include joint demosaicing and deblurring [125], joint denoising and deblurring [126], joint demosaicing and HDR reconstruction [101, 122], where all report reduction in artifacts.

Heide et al. [127] proposed an adaptive framework for multiple joint computational photography tasks such as denoising, color interpolation, and HDR reconstruction. Their method solved an inverse problem by combining different global natural image priors, such as block-matching and 3D filtering (BM3D), and total variational (TV). Although their framework was able to adapt for different tasks, including reconstruction of the interlaced HDR image, it is highly dependent on priors, such that if an image violates the priors, it can result in overly smoothed or sharpened edges. Furthermore, it does not utilize a heterogeneous camera noise model, and their method requires solving a global optimization problem, which is computationally expensive.

Kronander et al. [20] introduced a unified framework for the reconstruction of multi-exposure raw images captured with an HDR video camera shown in Figure 3.2. Their reconstruction method incorporates an accurate statistical noise model of the camera sensor, which was suggested earlier by [76]. This thesis proposes HDR reconstruction methods based on [20], where the reconstruction window is adapted for spatially multiplexed imaging systems. In the following section we describe the details of our novel HDR reconstruction algorithm.

3.4 Dual-ISO Spatial Multiplexing

Kernel regression (KR) is an effective tool for the super-resolution of an image from single or multiple frames, as well as reducing the noise that is introduced by the imaging pipeline [128]. Locally adaptive filtering has been used in a variety of image processing operations, such as normalized convolution [129], and bilateral filtering [130]. A deep connection between these methods and traditional non-parametric statistical methods [131] have been shown in [128, 132]. As discussed earlier, a unified framework that solves multiple tasks simultaneously can reduce the error of reconstruction. This section explains a reconstruction method that adopts a unified statistical model similar to [20] on a spatially interlaced image with two gain values. The final HDR image is accurately reconstructed in a joint process of resampling, color interpolation, and denoising, as shown in Figure 3.4. The accuracy of the reconstruction is based on including the statistical noise properties of the sensor in the reconstruction model. Different gain settings have different effects on the noise characteristics that must be considered in the statistical model.

The input samples in the local neighborhood of a reconstruction point are interpolated using local polynomial approximation in the method proposed in this section. The LPA combined with *maximum likelihood* is used for designing a nonlinear filter by fitting a polynomial in a sliding window. A desirable filter is defined by the window, its size, and the order of the polynomial. In this method, we employed a Gaussian kernel to interpolate the observed pixel values. The interpolation weights are calculated using the estimated variance of each sample point to ensure that reliable samples are weighted higher than noisy, and unreliable ones. The effect of the Gaussian filter is to guarantee that samples further away from the reconstruction point have lower weights compared to neighboring points. The size of the filtering kernel can be adjusted iteratively to the statistical characteristics of the signal and its underlying variance to preserve the edges and important features of the image. In what follows, the details of this algorithm are explained.

3.4.1 Sensor Noise Model

The number of photoelectrons, e_i , accumulated at each pixel, indexed by i , during the exposure time, t , follows a Poisson distribution where the expected value and variance are:

$$\begin{aligned} E[e_i] &= t(a_i f_i) \\ \text{Var}[e_i] &= E[e_i], \end{aligned} \tag{3.1}$$

where a_i is pixel non-uniformity (see 2.1.2). Each pixel value y_i , corresponds to the number of photo-induced electrons collected on the sensor per unit time, known as incident radiant power, f_i . The pixel values, y_i , consist of the incident radiant

power, f_i , and the measurement noise that is dependent on sensor characteristics such as the gain, g , and the signal independent readout noise:

$$y_i = g(e_i) + r_i(g), \quad (3.2)$$

where $r_i(g)$ is the readout noise that is dependent on the gain settings of the sensor. For simplicity, exposure settings longer than one second are not considered in this thesis. Therefore, since the effect of dark current noise is negligible for short exposure times, it is not included in the noise model, see 2.1.2. Previous methods consider a simple parametric model for the readout noise dependence on the gain [20, 71]. This model is not suitable for multiple gain settings found on modern cameras, and instead, the readout noise, $r_i(g)$, is estimated carefully for each gain/ISO setting through a calibration process, see Section 3.4.3.

Recovering the full HDR image requires us to transform the noisy digital pixel values, y_i , to an estimation of incident radiant power, $\hat{f}_i$, such that the pixel values are in the same radiometric space. The sensor gain varies for every pixel of the sensor, but the exposure scaling coefficient, exposure time, and other parameters are similar over the sensor. The pixel values are transformed using a radiometric model [9, 20, 133], where non-saturated pixels follow a random variable distribution as follows:

$$y_i \sim N(g_i a_i t f_i + \mu_R, g_i^2 a_i^2 t^2 f_i + \sigma_R^2(g_i)), \quad (3.3)$$

where g_i is the per-pixel gain, a_i is the pixel non-uniformity, t is the exposure time, and μ_R and σ_R^2 are the mean and the variance of the readout noise. The variance of the radiant power can be estimated through a calibrated radiometric model that will be explained in Section 3.4.3. We assume the readout noise follows a Gaussian distribution similar to [20, 116].

The incident radiant power from the noisy digital input sample values y_i is estimated using the following:

$$\hat{f}_i = \frac{y_i - b_i}{g_i t a_i}, \quad (3.4)$$

where b_i is the bias frame or black image computed as the average of a large set of black images captured carefully without any light rays reaching the sensor.

Other sensor noise models that include the effects of saturation on the variance of the pixels, $\text{Var}[y_i]$, can also be applied [134]. These models, however, are impractical and complicated for the local polynomial approximation that is used in this thesis since they consider y_i to follow a clipped normal distribution.

3.4.2 Variance Estimate

To calculate the variance of the estimated radiant power, $\hat{f}_i$, the assumption is that $\hat{f}_i$ has a normal distribution with mean f_i and standard deviation $\sigma_{\hat{f}}$. A normal

distribution can approximate the photon shot noise with a Poisson distribution in the bright regions of the image. In low light environments or darker regions of the image, the photon shot noise is dominated by signal-independent readout noise that can as well be approximated by a normal distribution. Assuming that the pixels are non-saturated, the variance of $\hat{f}_i$ can be found by:

$$Var[\hat{f}_i] = \sigma_{\hat{f}}^2 = \frac{g_i^2(Var[e_i]) + Var[r_i(g)]}{g_i^2 t^2 a_i^2}. \quad (3.5)$$

To estimate $Var[e_i]$, we need to know the actual value of f_i due to the existence of photon shot noise. However, in practice, the noisy estimate of the incident radiant power, $\hat{f}_i$, obtained from Equation (3.4), is used instead. Therefore, the variance of the estimated incident radiant power is similarly approximated as:

$$\hat{\sigma}_{\hat{f}_i}^2 = \frac{g_i^2 t a_i \hat{f}_i + \hat{\sigma}_R^2(g_i)}{g_i^2 t^2 a_i^2}, \quad (3.6)$$

where g_i , a_i and $\hat{\sigma}_R^2(g_i)$ are calculated through a calibration process. The estimated variances are assumed to be independent of each other, and the effect of pixel cross-talk is not considered here.

The assumption so far has been that there are no saturated pixels in the captured image. Nonetheless, in practice, some pixels might get saturated due to excessive amounts of light, e.g., direct sunlight. These saturated pixels are unreliable and should be excluded from the computations using a *saturation frame* to avoid further errors. The saturation frame indicates at what level each pixel saturates. The pixels are compared with this frame and excluded if they are saturated.

3.4.3 Camera Parameters Calibration

The noise characteristics of the camera are estimated through a calibration process that is performed once for each camera with the same optical elements (e.g., lens). The pixel non-uniformity, a_i , can be estimated from a set of flat-field images. For simplicity, we consider that a_i is constant for all pixels. The readout noise parameters, e.g., mean, μ_R , and the variance, σ_R^2 , are estimated by capturing a set of dark images, $\{b_i\}$, using the same camera settings, (i.e. gain, g , and exposure time, t), with the lens cap on, such that no light rays arrive at the sensor. Per-pixel sensor gain g_i is estimated with the following relation:

$$\frac{Var[y_i] - Var[b_i]}{E[y_i] - E[b_i]} = \frac{g_i^2 Var[e_i]}{g_i E[e_i]} = g_i. \quad (3.7)$$

As the number of collected photoelectrons, e_i , follows the Poisson distribution with expected variance $Var[e_i] = E[e_i]$, (see Equation (3.1)), the second equality

Figure 3.5: Mean standard deviation of the readout noise σ_R and its dependence on the gain settings, for Canon 5D Mark II and Canon 5D Mark III.

is simplified to g_i . Additionally, $E[y_i]$ is estimated from the flat-field images. Moreover, $E[b_i]$ and $\text{Var}[b_i]$ can be computed using the dark images, as explained above. Figure 3.5 illustrates the mean standard deviation of the readout noise when the ISO parameter is changed for two camera models: *Canon 5D Mark II* and *Canon 5D Mark III*.

3.4.4 Local Polynomial Approximation

To approximate an HDR value, $\tilde{f}_j$, at output HDR pixel location $\mathbf{X}_j = [x_1, x_2]$, the observation samples of the incident radiant power are fitted to a polynomial in its local neighborhood using an LPA [84] or kernel regression [128]. Since the radiant power f_j can be considered as a smooth function in a local neighborhood around the output location $\mathbf{X}_j$, we employ an M -th order Taylor series expansion to approximate the radiant power at an observation location $\mathbf{X}_k$ near the output location $\mathbf{X}_j$, as follows:

$$\begin{aligned} \tilde{f}(\mathbf{X}_k) = & C_0 + C_1(\mathbf{X}_k - \mathbf{X}_j) \\ & + C_2 \text{tril}\{(\mathbf{X}_k - \mathbf{X}_j)(\mathbf{X}_k - \mathbf{X}_j)^T\} + \dots, \end{aligned} \quad (3.8)$$

where *tril* lexicographically vectorizes the lower triangular part of a symmetric matrix ² and $C_{1:M}$ are the fitted polynomial coefficients defined as:

$$C_0 = f(\mathbf{X}_j) \quad (3.9)$$

$$C_1 = \nabla f(\mathbf{X}_j) = \left[\frac{\partial f(\mathbf{X}_j)}{\partial x_1}, \frac{\partial f(\mathbf{X}_j)}{\partial x_2} \right] \quad (3.10)$$

$$C_2 = \frac{1}{2} \left[\frac{\partial^2 f(\mathbf{X}_j)}{\partial x_1^2}, 2 \frac{\partial^2 f(\mathbf{X}_j)}{\partial x_1 \partial x_2}, \frac{\partial^2 f(\mathbf{X}_j)}{\partial x_2^2} \right]. \quad (3.11)$$

Therefore, the HDR pixel value, at the output location $\mathbf{X}_j$ is estimated by $C_0 = f(\mathbf{X}_j)$. Moreover, C_1 and C_2 are the first and second order gradients, respectively.

3.4.5 Maximum Localized Likelihood Fitting

The polynomial coefficients, $C_{1:M}$, are estimated by maximizing the localized likelihood function [135] that is defined by a smoothing window centered at $\mathbf{X}_j$:

$$\mathcal{W}_{\mathbf{H}}(\mathbf{X}_j) = \frac{1}{\det(\mathbf{H})\mathcal{W}(\mathbf{H}^{-1}\mathbf{X}_j)} \quad (3.12)$$

where $\mathbf{H}$ is a 2×2 smoothing matrix that controls the shape and size of the filtering window. One choice to determine the smoothing window is to define it as a truncated Gaussian window. As a result, Equation (3.12) using a Gaussian window becomes:

$$\mathcal{W}_{\mathbf{H}}(\mathbf{X}_j) = \frac{1}{2\pi\mathbf{H}^2} \exp \left\{ \frac{-(\mathbf{X}_k - \mathbf{X}_j)^T (\mathbf{X}_k - \mathbf{X}_j)}{\mathbf{H}} \right\}. \quad (3.13)$$

The smoothing matrix, $\mathbf{H}$, affects the shape of the window function. Some approaches define a complex shape that adapts to the signal by rotation, elongation, and scaling of the window [128]. The simplest choice is to set $\mathbf{H} = h\mathbf{I}$, where $\mathbf{I}$ is the identity matrix and h is the global scale parameter that determines the size of the kernel. Setting this parameter to a fixed user-defined value will create a globally fixed-size sliding window, as presented in *Paper A*. However, this parameter can be set for each reconstructed pixel such that the filter kernel adapts to the statistical features of the image as will be explained in Section 3.4.6.

The observed pixel samples in the local neighborhood of pixel location $\mathbf{X}_j$, are denoted as $\hat{f}_k$ with linear index $k = 1 \dots K$. By assuming that the radiant power estimates, $\hat{f}_k$, obtained by Equation (3.4), have a normal distribution, we maximize the localized likelihood function to approximate the polynomial coefficients $\tilde{C}$ by the weighted least squares estimate:

² i.e. $\text{tril} \left(\begin{bmatrix} a & b & c \\ b & e & f \\ c & f & i \end{bmatrix} \right) = [a, b, c, e, f, i]^T$

Figure 3.6: Unified HDR reconstruction results of real data from Canon 5D MarkIII with alternating ISO readout. Our method with dual-ISO settings: 100-800, 100-1600 and Magic Lantern (ML) DUALISO method [114] with dual-ISO settings: 100-800.

$$\tilde{C} = (\Phi^T \mathbf{W} \Phi)^{-1} \Phi^T \mathbf{W} \bar{f}, \quad (3.14)$$

where

$$\begin{aligned} \bar{f} &= [\hat{f}_1, \hat{f}_2, \dots, \hat{f}_K]^T \\ W &= \text{diag}\left[\frac{\mathcal{W}_h(\mathbf{X}_1)}{\hat{\sigma}_{\hat{f}_1}^2}, \frac{\mathcal{W}_h(\mathbf{X}_2)}{\hat{\sigma}_{\hat{f}_2}^2}, \dots, \frac{\mathcal{W}_h(\mathbf{X}_K)}{\hat{\sigma}_{\hat{f}_K}^2}\right] \\ \Phi &= \begin{bmatrix} 1 & (\mathbf{X}_1 - \mathbf{X}_j) & \text{tril}^T\{(\mathbf{X}_1 - \mathbf{X}_j)(\mathbf{X}_1 - \mathbf{X}_j)^T\} & \dots \\ 1 & (\mathbf{X}_2 - \mathbf{X}_j) & \text{tril}^T\{(\mathbf{X}_2 - \mathbf{X}_j)(\mathbf{X}_2 - \mathbf{X}_j)^T\} & \dots \\ \vdots & \vdots & \vdots & \vdots \\ 1 & (\mathbf{X}_K - \mathbf{X}_j) & \text{tril}^T\{(\mathbf{X}_K - \mathbf{X}_j)(\mathbf{X}_K - \mathbf{X}_j)^T\} & \dots \end{bmatrix}. \end{aligned} \quad (3.15)$$

This approach allows us to estimate the polynomial coefficients $C_{1:M}$ effectively for an M -th order polynomial and to compute the HDR pixel value $\hat{f}_j$ for a given h value. The smoothing parameter, h , determines the trade-off between the bias and the variance of the estimation, which corresponds to the image sharpness vs. noise reduction. Figure 3.6 illustrates the result of applying the kernel regression with a globally fixed scaling parameter, h , using the proposed method. The figure shows a comparison of our approach with ISO 100-800 and ISO 100-1600 and Magic Lantern DUALISO method [114] with ISO 100-800.

Figure 3.7: *The effect of the kernel size on the captured signal and the edge information.*

3.4.6 Adaptive Kernel Regression

As mentioned earlier, the size of the kernel function determines a trade-off between the variance and the bias of the reconstructed image. Large window size reduces the noise, and hence the variance, while increasing the bias, e.g. by smoothing out sharp edges. The window size, ideally, should be adjusted to the underlying signal to preserve important features of the image, such as edges, while removing noise in homogeneous regions.

The smoothing parameter, h , determines the size of the smoothing kernel. The effect of the smoothing parameter is illustrated in Figure 3.7, where the signal value, the black point in the figure, is estimated by a kernel with three sizes: $0 < h_0 < h_1 < h_2$. When incrementing the kernel size from h_0 to h_1 , the variance of the estimated value can still be explained by the variance of the underlying signal. However, increasing the window size too much as in h_2 forces the kernel to cross the edge, where the estimated point is no longer explained by the signal variance, resulting in over-smoothness. The size of the filter kernel can be adapted iteratively to increase gradually until a certain criterion is achieved. Selecting the filter size adaptively is a necessary element of a successful local approximation [84].

Algorithm 1 explains each step that is required to reconstruct an output HDR pixel f_j by adapting the smoothing parameter h_l . In each iteration of the algorithm, the HDR signal and its variance are estimated for the current smoothing parameter. Afterwards, an update rule, explained in the next section, is applied that determines whether the current parameters satisfy the estimation or not. The iteration continues until no further update is valid, or the maximum value of the smoothing parameter, h_{max} (defined by the user), is reached. We introduce two updating rules in the next two sections and explain how the variance of the estimated signal is computed.

Input: Bayer dual-ISO image, $y_i, b_i, h_{min}, h_{max}$.
Output: Reconstructed HDR image

```

1 for each color channel in  $R, G, B$  do
2   for each HDR pixel estimate  $\tilde{f}_j$  do
3      $h_l \leftarrow h_{min}$ ;
4     Compute  $\tilde{f}_{j,h_l}$  using LPA with degree  $M$ ;
5      $\tilde{f}_j \leftarrow \tilde{f}_{j,h_l}$ ;
6     while  $h_l < h_{max}$  do
7        $h_{l-1} \leftarrow h_l$  and  $h_l \leftarrow h_{l-1} + h_{inc}$ ;
8       Estimate  $\tilde{f}_{j,h_l}$  using LPA with degree  $M$ ;
9       Estimate  $\tilde{\sigma}_{\tilde{f}_{j,h_l}}$  and the reconstruction error  $\epsilon_l$  (for EVS);
10      Apply update rule (ICI or EVS) based on  $\tilde{f}_{j,h_l}$ ;
11      if variation in  $\tilde{f}_{j,h_l}$  can be explained by  $\tilde{\sigma}_{\tilde{f}_{j,h_l}}$  then
12        |  $\tilde{f}_j \leftarrow \tilde{f}_{j,h_l}$ ;
13      end
14      else
15        | Break;
16      end
17    end
18  end
19 end

```

Algorithm 1: Adaptive HDR reconstruction.

3.4.7 Update Rule 1: Error of Estimation Versus Standard Deviation (EVS)

This updating rule is built upon the intuition to examine whether the polynomial model is a good fit for the underlying signal or not. A good fit is found when the difference between the weighted mean reconstruction error is explained by the expected signal variations, i.e. the weighted mean and standard deviation. The noise can affect the capabilities of the polynomial model in representing the data. Therefore, to reduce this effect, in each iteration, l , of Algorithm 1, the smoothing parameter, h_l , is increased with an offset, h_{inc} , that is controlled by the user. The EVS update rule estimates the weighted reconstruction error, ϵ_l as follows:

$$\epsilon_l = \sqrt{\sum_{k=1}^K \mathbf{W}^2(k, k) (\tilde{f}(\mathbf{X}_k) - \hat{f}_k)^2}, \quad (3.16)$$

Figure 3.8: *Reconstruction process of a simulated dual-ISO image. Top left shows the raw input image with the CFA Bayer pattern and dual-ISO row pattern. Top right indicates the tone mapped HDR reconstructed image with the EVS updating rule. Bottom rows extracted images from left to right: cutout of the raw image, window size for R,G,B color channels with $\Gamma = 1.0$, and the cutout of the reconstructed HDR image side by side the reference.*

where K is the number of pixels in the neighborhood, and $\mathbf{W}$ is the weight matrix considering both the variance of the original pixels and the spatial Gaussian kernel as described in Equation (3.15). The error of reconstruction, ϵ_l , is compared to the weighted standard deviation of the HDR pixel estimate that is obtained from the covariance matrix $\mathbf{M}_{\tilde{C}}$ for the fitted polynomial coefficients, $\tilde{C}$, given by:

$$\mathbf{M}_{\tilde{C}} = (\Phi^T \mathbf{W} \Phi)^{-1} \Phi^T \mathbf{W} \Sigma \mathbf{W}^T \Phi (\Phi^T \mathbf{W}^T \Phi)^{-1}, \quad (3.17)$$

where $\Sigma = \text{diag}[\sigma_{\hat{f}_1}^2, \sigma_{\hat{f}_2}^2, \dots, \sigma_{\hat{f}_k}^2]$ is the variance of the observations. In each iteration, the smoothing parameter, h_l is updated to $h_l = h_{l-1} + h_{inc}$ as long as the reconstruction error is smaller than the standard deviation, $\epsilon_l < \Gamma \tilde{\sigma}_{\tilde{z}_j, h_l}$, where Γ is a constant defined by the user to control the trade-off between the level of denoising that is applied by the kernel function and the reconstruction quality.

Figure 3.8 illustrates the EVS update rule and how it fits the structure of the image and preserves the edges by keeping the kernel size small around those areas. The effect of the algorithm on different color channels is also shown in this figure.

Figure 3.9: Reconstruction of a simulated dual-ISO scene with different methods for comparison: (a) LPA, $M = 2$, from left to right: $h = 0.6, 1.4$, and 5.0 ; (b) SKR [128], $M = 2$ from left to right $h = 0.6, 1.4$, and 5.0 ; (c) our method with ICI, $M = 2$ from left to right $\Gamma = 0.6, 1.0$, and 1.4 ; (d) our method with EVS, $M = 2$ from left to right $\Gamma = 0.6, 1.0$, and 1.4 .

Algorithm		PU-SSIM	Log-Normalized-SSIM	HDR-VDP 2.2
LPA	$h = 0.6$	0.9737	0.9883	54.3042
	$h = 1.4$	0.9893	0.9903	56.8246
	$h = 5.0$	0.9683	0.9828	53.5847
SKR	$h = 0.6$	0.9655	0.9766	51.0470
	$h = 1.4$	0.9912	0.9924	56.9265
	$h = 5.0$	0.9826	0.9897	55.3868
ICI	$\Gamma = 0.6$	0.9893	0.9929	57.5614
	$\Gamma = 1.0$	0.9924	0.9937	59.3245
	$\Gamma = 1.4$	0.9928	0.9937	59.8197
EVS	$\Gamma = 0.6$	0.9772	0.9893	54.4240
	$\Gamma = 1.0$	0.9930	0.9935	58.8586
	$\Gamma = 1.4$	0.9945	0.9943	58.4647

Table 3.1: Quantitative comparisons of Figure 3.9.

Figure 3.10: *Reconstruction of a dual-ISO image taken with Canon 5D Mark III using the ICI update rule with $M = 2$, $h \in [0.6, 5.0]$ and $\Gamma = 1.0$ with ISO settings: 100-1600.*

3.4.8 Update Rule 2: Intersection of Confidence Intervals (ICI)

Another way to adapt the smoothing parameter to the signal is based on the ICI algorithm [84]. This algorithm ensures that the largest scaling parameter, $h_{min} \leq h_l \leq h_{max}$, is obtained in a local neighborhood of the estimation point such that the polynomial model would stay as a possible fit to the signal. For each iteration l of Algorithm 1, the ICI rule indicates a confidence interval, $D_l = [L_l, U_l]$, based on the estimated radiant power value, $\tilde{f}_{j,h_l}(x)$ for the current scaling parameter,

h_l , as follows:

$$L_l = \tilde{f}_{j,h_l}(x) - \Gamma \tilde{\sigma}_{z_j,h_i}, \quad (3.18)$$

$$U_l = \tilde{f}_{j,h_l}(x) + \Gamma \tilde{\sigma}_{z_j,h_i}, \quad (3.19)$$

where, $\tilde{\sigma}_{z_j,h_i}$ is the weighted standard deviation of this estimation calculated using Equation 3.17. The user-defined parameter, Γ , controls the length of the intersection interval. During each iteration, h_l is updated to $h_l = h_{l-1} + h_{inc}$ if there is an overlap between the confidence intervals, D_{l-1} and D_l . In practice, Γ creates a trade-off between the noise removal and the sharpness of the image (to read more about ICI rule see [136]).

A comparison of the proposed fixed and adaptive kernel methods with the two updating rules and the steering kernel regression (SKR) method [128] are demonstrated in Figure 3.9 and Table 3.1. The reconstruction is evaluated using different parameters that are mentioned in this section. Both proposed updating rules outperform the other methods in denoising and image sharpness. The adaptive window support can reduce the color artifacts around the sharp edges. Although SKR adapts the shape of the kernel to the features of the image, it fails to recover areas where both gain settings are saturated. Increasing the size of the kernel improves this problem but leads to over-blurring the image. On the contrary, our proposed methods with both ICI and EVS updating rules reconstruct the saturated regions with minimum artifacts. As shown in Table 3.1, ICI is less dependent on the Γ coefficient value compared to EVS update rule.

Figure 3.10 shows the result of reconstructing a real dual-ISO image taken by Canon 5D Mark III with dual-ISO settings 100 – 1600, ($g = 0.3 - 4.8$), using the ICI update rule where a high-quality HDR image is achieved while preserving the details.

3.4.9 Spatially Interlaced Gain Patterns

As explained earlier, different patterns affect the reconstructed HDR image. Regular patterns such as the row pattern of dual-ISO mode on Canon Cameras (with Magic Lantern firmware), introduce aliasing and color artifacts, particularly at high-frequency edges. We evaluated our method on two other spatially multiplexed dual gain patterns using simulation, as shown in Figure 3.11. The results of these patterns together with row pattern of the Magic Lantern dual-ISO are shown at the left side of Figure 3.11, where the aliasing artifacts are significantly improved by the diagonal or block pattern. This is confirmed in Table 3.2 which shows that the diagonal pattern results in a better reconstruction quality in both PU-SSIM [80] and HDR-VDP-2 [137] quality metrics.

Figure 3.11: Comparison of different gain patterns: Block, diagonal, and row patterns. EVS adaptation rule with $M = 2, \Gamma = 1.0$.

Gain Pattern	Diagonal Pattern	Block Pattern	Row Pattern
PU-SSIM	0.9967	0.9966	0.9960
HDR VDP v2.2	66.21	66.09	62.62

Table 3.2: Quantitative comparisons for gain patterns: diagonal, block, row, as shown in Figure 3.11.

3.5 Summary and Future Work

This chapter presented a novel unified HDR reconstruction algorithm based on the statistical model of the sensor noise for the spatially interlaced images with variable exposures. The local polynomial model, based on the non-parametric regression, estimates the HDR pixel value in a noisy neighborhood around the pixel by applying a localized maximum likelihood approach. Two updating rules for adapting the filter size to the features of the signal and noise were presented. Adaptive kernel windows show improvements in image denoising and reconstruction quality while reducing color artifacts. The proposed framework is applied to both synthetically created data and real images captured by a conventional camera with spatially varying gain settings.

The future research direction involves developing the reconstruction framework for the HDR videos captured with multiple gain settings. A three-dimensional kernel window must be considered that can adapt to both spatial and temporal domains to include observation points within each frame and across multiple frames of the video, similar to the method of [138]. Another intriguing research direction is to consider cross channel correlation in the reconstruction model to improve the

accuracy of demosaicing and reducing the color artifacts.

Our proposed reconstruction method adapts the kernel size to the image content and noise model. Combining our approach with the idea from steering kernel regression (SKR) [128] where the shape of the kernel is changed based on the important features in the image can further improve the result.

Sparse Representation and Compressed Sensing

This chapter describes the basic concepts used in the second part of this thesis that relate to the sparse light field imaging techniques. The covered topics include concepts such as sparsity and sampling, dictionary learning, overcomplete representation and redundancy, compressed sensing, and sparse representations. This chapter provides the theoretical background for compressive light field imaging, light field sparse representation, encoding, and rendering that are presented in chapters 5 and 6.

4.1 Sparse Signal Representation

The field of signal processing has seen significant advancements on efficient modeling and sampling of signals during the last two decades with the introduction of *sparse representation* of signals. Many applications in this field benefit from *sparsity*, such as restoration, feature extraction, and compression. The attractiveness of sparse representation techniques comes from their capacity to represent the data compactly. A signal $\mathbf{x}$ is considered sparse if most of its entries are zero. A τ -sparse signal is defined as $\|\mathbf{x}\|_0 < \tau$ where the ℓ_0 pseudo-norm defines the number of nonzero elements in the signal $\mathbf{x}$. A non-sparse signal can be *sparsified* if it is represented in a suitable transform domain. Signal $\mathbf{x} \in \mathbb{R}^N$ can be represented as a superposition of elements of a *dictionary* $\mathbf{D} \in \mathbb{R}^{N \times T}$ using a vector of *coefficients*

$\mathbf{s} \in \mathbb{R}^T$ as follows:

$$\mathbf{x} = \mathbf{D}\mathbf{s}. \quad (4.1)$$

The elements of a dictionary are called *atoms*. A few examples of these atoms include wavelets, monomials, and sinusoids. Linear superposition of multiple atoms in a building block can create a complex waveform. Given a dictionary, through an *encoding* process, each signal $\mathbf{x}$ is projected onto the dictionary to obtain a vector of coefficients as: $\mathbf{s} = \mathbf{D}^T \mathbf{x}$. The *decoding* phase, on the other hand, reconstructs the signal by superposing atoms using the coefficient vector, Equation (4.1).

An *overcomplete* or redundant dictionary has more columns than rows, $T > N$. Similar to dictionaries in natural language processing, the richer and larger the dictionary, the better it can represent the signal. An overcomplete dictionary with high redundancy enhances the sparse representation in the signal and improves tasks such as pattern recognition, compression, and compressive sensing [139]. On the other hand, there is no guarantee for obtaining a unique solution from the sparse coefficient vector associated with a dictionary that exhibits high redundancy. This is due to the fact that the linear system to be solved, i.e. Equation 4.1, is under-determined.

To obtain a suitable representation, first, it is required to decompose the signal into its primary components, i.e. dictionaries and coefficients. Secondly, such decompositions need to adapt to the signal properties. Wavelets are suitable for representing isotropic features, while overcomplete dictionaries such as ridgelet [140] and curvelets [141] lead to sparser representation for anisotropic and curvilinear structures. For periodic and stationary signals, the Fourier basis is well-suited [142]. Overcomplete dictionaries are typically preferable if there is a fast algorithm to transform the signal to and from the dictionary space. Choosing one transform rather than another requires prior information on the structure of the data. When such information is not available, one can learn the dictionary.

Finding a suitable dictionary that admits sparse representation of the data is a cumbersome task. To solve this problem, *Dictionary Learning* (DL) has emerged to create the possibility of adapting a dictionary $\mathbf{D}$ to the data through a set of examples that represent such data, known as the training set. Research has shown that data-driven dictionaries lead to higher performance when compared to pre-designed analytical dictionaries in applications such as inpainting [143], denoising [14], and other types of inverse problems [144]. The next section explains the dictionary learning for sparse representation of data.

4.2 Sparse Dictionary Learning

To represent a signal using the sparse domain of a learned dictionary requires solving the following:

$$\min_{\mathbf{s}} \|\mathbf{s}\|_0 \quad s.t. \quad \|\mathbf{y} - \mathbf{D}\mathbf{s}\|_2^2 \leq \epsilon, \quad (4.2)$$

where ϵ can be set as an estimation for noise. If we have an estimate for the sparsity value, τ , we can set the constraint to $\|\mathbf{s}\|_0 \leq \tau$. However, in the absence of these estimates, Equation (4.2) can be calculated as an unconstrained problem known as the basis pursuit denoising (BPDN) [145]:

$$\min_{\mathbf{s}} \lambda \|\mathbf{s}\|_0 + \frac{1}{2} \|\mathbf{y} - \mathbf{D}\mathbf{s}\|_2^2. \quad (4.3)$$

This problem is NP-hard due to the non-convexity of the ℓ_0 norm, which can be replaced by an ℓ_1 norm to create a convex optimization problem with a global optima. For more insights into the geometrical interpretation of the sparse recovery problem, see [146, 147, 148].

There are different categories of approaches for estimating a sparse signal. For instance, there exists heuristic methods using iterative approaches such as *Matching Pursuit* (MP) [149] or *Orthogonal Matching Pursuit* (OMP) [150]. Other methods based on optimization, and relaxation of the ℓ_0 norm with ℓ_1 norm, attempt to solve a convex problem using linear programming [151, 152, 153, 154], see [155] for a comprehensive survey. Although there is a convergence guarantee for these methods, they have a higher computational complexity compared to greedy algorithms. Deriving theoretical guarantees for the heuristic methods often requires stringent conditions compared to ℓ_1 methods [156].

Another category of algorithms is called *hybrid* methods that try to solve the non-convex ℓ_0 sparse recovery problem by using surrogate functions, such as *smoothly clipped absolute deviation* (SCAD) norm [157], see also [158, 159]. Another example is the SL0 algorithm [160], which is, in some cases, two to three orders of magnitude faster than other linear programming methods while maintaining the accuracy.

So far, we have considered one-dimensional signals. The sparse representation can also be applied to multidimensional signals. When the dimensionality of data increases, the self-similarities or *coherence* within it also grows [161]. Representing the signal in a sparse domain where it can be described with a few coefficients can lead to faster processing and efficient storage of the data. Reducing the dimensionality of a signal has been studied through low-rank approximation techniques [162] and nonlinear methods known as manifold learning [163, 164] based on the assumption that the high dimensional data lies on a low dimensional manifold.

Representing the data in the sparse domain is not always feasible. One method to enforce sparsity is to decompose the signal into a dictionary and a set of coefficients, followed by the nullification of small or non-significant coefficients. A popular method for decomposing a matrix is singular value decomposition (SVD) that factorizes a matrix $\mathbf{M} \in \mathbb{R}^{m \times n}$ to:

$$\mathbf{M} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T, \quad (4.4)$$

where $\mathbf{U} \in \mathbb{R}^{m \times m}$ and $\mathbf{V} \in \mathbb{R}^{n \times n}$ are orthogonal matrices containing singular vectors and $\mathbf{\Sigma} \in \mathbb{R}^{m \times n}$ is a diagonal matrix containing singular values in a descending order. The rank of the matrix is defined by the number of nonzero singular values from SVD. Moreover, one can approximate the matrix $\mathbf{M}$ by setting the smallest singular values to zero. Such a decomposition is known as the truncated-SVD.

Similar to one-dimensional signals, a high dimensional signal, also known as tensor, can also be decomposed using dictionaries constructed by Fourier, wavelets, or learning-based basis functions [165]. For instance one of the most commonly used tensor decomposition methods is higher-order SVD (HOSVD) [166] defined for a high dimensional signal as:

$$\mathcal{L} = \mathcal{S} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \cdots \times_n \mathbf{U}^{(n)}, \quad (4.5)$$

where $\{\mathbf{U}^{(i)}\}_{i=1}^n$ are orthogonal or orthonormal matrices and $\mathcal{S}$ is the core tensor. It is possible to compress $\mathcal{L}$ by setting the small values in $\mathcal{S}$ to zero, as well as discarding the columns of $\{\mathbf{U}^{(i)}\}_{i=1}^n$ that correspond to the nullified elements; this algorithm is known as truncated-HOSVD. Note that unlike truncated-SVD, the approximation obtained from the truncated-HOSVD is not necessarily optimal in an ℓ_2 sense.

4.2.1 Patch-Based Learning

Training a dictionary on a large number of exemplars is numerically intractable. Natural images exhibit nonlocal self-similarities, where blocks or patches from the image share similar content. Therefore, instead of the whole image, we can employ image patches for learning a dictionary [14, 143, 144]. Since the size of the dictionary atoms is equal to the signal size, using image patches reduces the size of the atoms, and hence the computational complexity. The other benefit of patch-based learning is that one can obtain from a limited set of images, or other types of data, a large number of exemplars for training.

4.2.2 Examples of Dictionary Training Algorithms

K-SVD is a learning-based algorithm that generalizes the k-means clustering method. The method is based on updating the coefficient vectors by sparse coding

using the current dictionary and fixing the coefficients while altering the atoms of the dictionary for a better fit to the data [165, 167]. The name K-SVD comes from the Singular Value Decomposing (SVD) algorithm that forms the core of the dictionary update step. The process is repeated until all atoms are updated. K-SVD ensures that the elements of the input data are encoded by a linear combination of τ elements:

$$\arg \min_{\mathbf{D}, \mathbf{s}} \|\mathbf{x} - \mathbf{D}\mathbf{s}\|_2^2 \quad s.t. \quad \|\mathbf{s}\|_0 \leq \tau, \quad (4.6)$$

where τ denotes the sparsity value.

K-SVD has applications in many research areas such as image and audio processing, document analysis, and biology. As many algorithms, K-SVD has a few disadvantages; for instance, getting stuck at local minima due to the non-convexity of the problem. The computational complexity of training a K-SVD dictionary is high, which makes this method only suitable for representing small signals or small patches from the data.

Online dictionary learning for sparse coding Most algorithms assume that all training data is available at the time of learning the dictionary. However, in some cases, it is not possible to fit large amounts of data into memory, or it is required that the input data is streamed gradually to the algorithm. To resolve these issues, a series of algorithms have been developed to iteratively update the sparse model as soon as new data is available [143]. In these methods, first, the dictionary is initialized by random values. Then with a fixed dictionary, sparse coefficient vectors are calculated by solving:

$$\mathbf{s}_t = \arg \min_{\mathbf{s}} \frac{1}{2} \|\mathbf{x}_t - \mathbf{D}_{t-1}\mathbf{s}\|_2^2 + \lambda \|\mathbf{s}\|_1. \quad (4.7)$$

Once the coefficient vector, $\mathbf{s}$, is calculated, the dictionaries are updated by solving:

$$\mathbf{D}_t = \arg \min_{\mathbf{D}} \frac{1}{t} \sum_{i=1}^t \left(\frac{1}{2} \|\mathbf{x}_i - \mathbf{D}\mathbf{s}_i\|_2^2 + \lambda \|\mathbf{s}_i\|_1 \right), \quad (4.8)$$

where t is the t -th input sample. Experiments show that using online dictionary learning for large training sets leads to faster dictionary learning when compared to the other methods.

4.3 Compressive Sensing

Many problems in science and technology require reconstructing a signal from its measurements. Assuming a linear sampling process, this corresponds to solving:

$$\mathbf{y} = \Phi \mathbf{x}, \quad (4.9)$$

Figure 4.1: (a) Original image (b) Reconstruction using 10% of the largest absolute Haar wavelet coefficients where the rest of the coefficients are set to zero.

where $\mathbf{y} \in \mathbb{R}^m$ is the measured signal, $\Phi \in \mathbb{R}^{m \times N}$ is a matrix that models the linear measurement process, and $\mathbf{x} \in \mathbb{R}^N$ is the original signal we would like to reconstruct. Since $m < N$, the linear system is under-determined, and there exist infinitely many solutions to it, which means it is impossible to recover the signal without additional information. The Shannon-Nyquist sampling theorem states in order to ensure full recovery of the signal, the sampling rate of a continuous-time signal must be twice its highest frequency. The introduction of *Compressive Sensing* (CS) [168] has shown that under certain assumptions, it is actually possible to reconstruct the signal from m *incoherent* measurements where m is much smaller than the signal length N . The number of required samples is related to the sparsity level of the signal. Incoherency means that the data information is spread out in the domain that represents it. A signal can be recovered perfectly from $\mathcal{O}(\tau \log N)$ incoherent measurements, which are much less than the original signal.

Many compression algorithms such as JPEG, MPEG [169] are developed based on the assumption of the sparsity of the signal. JPEG, for instance, employs discrete cosine transform (DCT) or wavelet basis, and by setting small coefficients to zero, it induces sparsity on the signal for compression. Figure 4.1 shows the sparsity that a natural image exhibits when transformed to a *Haar* wavelet basis. The reconstructed image is achieved by setting 90% of the Haar coefficients to zero, and as it is shown, the difference between the reconstructed and the original image is negligible. The main idea of compressive sensing is that since most signals are compressible under a certain basis function, far fewer samples are required in the acquisition process rather than applying the compression after sampling. In other words:

“We would like to compressively sense a compressible signal! This constitutes the basis goal of compressive sensing”[170].

Figure 4.2: Single-pixel camera representation. Image courtesy of Rice University [177].

Two components should be considered to achieve compressive sampling: The design of the measurement matrix Φ and an algorithm to find a solution to Equation (4.9). It has been shown that using random matrices with normal or Bernoulli i.i.d. distribution can uniquely recover $\mathbf{x}$ in Equation (4.9) [171, 172, 173]. It is also possible to find the optimal Φ using learning-based methods, which is an ongoing research problem [174, 175, 176].

As explained earlier, a sparse signal can be represented by linear combinations of dictionary atoms, such that $\mathbf{x} = \mathbf{D}\mathbf{s}$. Therefore, by substituting the signal $\mathbf{x}$ with its sparse representation in Equation (4.9), the sensing model can be rewritten as:

$$\mathbf{y} = \Phi \mathbf{D} \mathbf{s}. \quad (4.10)$$

Consequently, recovering a sparse signal requires solving the following:

$$\hat{\mathbf{s}} = \arg \min_{\mathbf{s}} \|\mathbf{s}\|_0 \quad \text{s.t.} \quad \|\mathbf{y} - \Phi \mathbf{D} \mathbf{s}\|_2^2 \leq \epsilon. \quad (4.11)$$

However, as we saw in the previous section, ℓ_0 -minimization is NP-hard, usually basis pursuit or ℓ_1 -minimization is used to solve the problem as:

$$\hat{\mathbf{s}} = \arg \min_{\mathbf{s}} \|\mathbf{s}\|_1 \quad \text{s.t.} \quad \|\mathbf{y} - \Phi \mathbf{D} \mathbf{s}\|_2^2 \leq \epsilon, \quad (4.12)$$

where the sensing matrix $\Phi \in \mathbb{R}^{p \times m}$ maps the signal of dimension m to a vector of dimension p . The dictionary $\mathbf{D} \in \mathbb{R}^{m \times k}$ enables the sparse representation of an unknown signal $\mathbf{x}$ with assumed sparsity τ . The number of measurements, p , are usually set by the user. The signal $\mathbf{x}$ can be recovered by computing $\hat{\mathbf{x}} = \mathbf{D}\hat{\mathbf{s}}$. All the algorithms discussed in Section 4.2, can be used to solve Equations (4.11) and (4.12).

Compressed sensing has applications in several fields, such as computational photography, computer vision, signal processing, medical imaging, and biology. An

early example of compressed sensing in imaging is the development of *Single-Pixel* cameras [178] that comprises a single point detector coupled with a spatial light modulator, i.e. DMD and a random number generator that creates a random pattern for DMD. The light reflected off the DMD is integrated into a single-pixel sensor. A reasonable quality image is achieved by capturing about 1300 measurements from the scene. The required amount of measurements is still much lower than the Nyquist rate (about 50 times). A diagram demonstrating this camera is shown in Figure 4.2. Other imaging systems that are formed with the help of compressed sensing include compressive light field camera [22, 23, 179], lensless imaging [180], and compressive video sensing [181, 182].

Besides imaging of visible light, CS has applications in areas where the invisible light needs to be captured and highly expensive sensor technologies are required. A few examples of such areas include capturing infrared light [183], magnetic resonance imaging (MRI) [184, 185], Computed Tomography (CT) imaging [186], bio-sensing for DNA-based identification [187], and compressive imaging [188].

In the next two chapters, the theoretical backgrounds presented here are employed to develop a framework for efficient light field encoding and rendering in Chapter 5 and designing a sensing model for a light field video camera in Chapter 6.

Light Field Compression and Rendering

In conventional imaging systems, the sensor integrates the incoming light rays from various angles towards a pixel, or photo-sensor, while the shutter is open. Therefore, the angular information of the incoming light is lost in this process. Light field imaging has emerged to capture both the spatial and angular information of light rays emanating from a scene towards the camera. During the past few decades, there have been several approaches to record the scene's illumination from various viewing angles and positions in space. The proposed techniques range from handheld cameras to bulky and expensive multi-camera systems. Light field imaging faces three main challenges: capturing a large amount of data corresponding to angular and spatial visual information of the scene (in real-time), efficiently storing the massive amounts of acquired information, and displaying the light field in real-time and according to the requirements of the application at hand. The focus of this chapter is to develop a framework to solve these challenges for light field images and videos.

5.1 Outline and Contributions

This chapter introduces a framework for sparse representation of light field images and videos. The framework develops a compression method based on the concept of sparse signal representation (introduced in Chapter 4) to efficiently and effectively encode such high dimensional data. The proposed framework provides a unified

Figure 5.1: Plenoptic function is a 7-dimensional function that measures the full luminance of the scene from every point of view (x, y, z) , every angle (θ, ϕ) , over different spectrum of light λ , and at each time step t .

solution by exploring the redundancies that exist within the light field data. A dictionary learning method is introduced that considers multi-dimensionality and redundancy of light fields by training a *Multidimensional Dictionary Ensemble* (MDE). The sparsity of the representation is increased by a novel pre-clustering method that improves the training time and reconstruction quality and leads to the construction of an *Aggregated Multidimensional Dictionary Ensemble* (AMDE). The compressed data with AMDE can be decoded and displayed efficiently since the method provides local access to the elements of the light field data. *Paper C* presents the full details and results of the framework that can be applied to any high dimensional visual data.

The trained AMDE is further improved in *Paper D* by a novel dictionary pruning algorithm based on pairwise mutual coherence of dictionaries in the ensemble. The pruned dictionary improves the encoding time significantly. Moreover, an extensive analysis of the effect of image noise on AMDE training and testing has been carried out, which shows that with a slight amount of denoising, the compression efficiency improves significantly. The sparse coefficients obtained from encoding, are used to reconstruct the light field in real-time with novel view generation.

The sparse coefficients can be further compressed by quantization and entropy coding, as presented in *Paper E*. An AMDE is trained for spherical inward-looking light field data, and the obtained sparse coefficients are quantized and encoded using Huffman coding. Our results show about three times improvements in the compression ratio while preserving the reconstruction quality.

In what follows, we describe the contributions mentioned above in more details. Specifically, in Section 5.2, we describe the plenoptic function and its relation to light fields. In Section 5.3, we describe how the light field is patched into small data points to be used for dictionary training. A framework for compression and rendering using MDE is presented with details in Section 5.4, followed by Section

Figure 5.2: *The two plane parameterization of the light fields.*

5.5, which explains the formation of the AMDE for designing a unified compression and rendering framework. Section 5.6 demonstrates the capabilities of the AMDE framework for compressing and rendering spherical light field data sets. Finally, a summary and possible future research directions are presented in Section 5.7.

5.2 Plenoptic Function

Light rays in a scene are described by a 7 dimensional (7D) function called the *plenoptic function* [45] that models a 3D dynamic environment at every point in space towards every possible direction over any range of wavelengths and at any given time:

$$l(x, y, z, \theta, \phi, \lambda, t), \quad (5.1)$$

where spatial coordinates are parameterized by (x, y, z) , angular coordinates by (θ, ϕ) , spectral domain by λ , and the time domain by t . The plenoptic term was derived from *plenus* meaning complete and *optus* meaning view. A sketch of the plenoptic function is shown in Figure 5.1. One can assume that light travels constantly through the air to simplify Equation (5.1) to a 6D function. By considering only a static scene, this function can be simplified to 5D, and by restricting the wavelength to RGB channels, and limiting light rays to the ones leaving the convex hull of the bounding box of the object, it is reduced to a 4D function [189, 190].

The plenoptic function can be described by any arbitrary surface that surrounds the environment. Thus, it can be parameterized in different ways, such as the two plane parameterizations, $l(x, y, s, t)$, where (x, y) defines the "*focal plane*" and (s, t) defines the "*aperture plane*", see Figure 5.2. Other parameterizations of light fields include spherical parameterization [191, 192], sphere-plane parameterization [192], and direction and point parameterization [193]; see [24] for a review of different parameterization methods.

Adopting the two plane parameterization for 6D light field video representation leads to the following formulation of the plenoptic function, Equation (5.1):

$$l(x, y, \theta, \phi, \lambda, t). \quad (5.2)$$

5.3 Light Field Data Points

Since processing a light field sequence is computationally infeasible, it is divided into smaller elements, called *data points* with the dimensionality $m_1 \times m_2 \times m_3 \times m_4 \times m_5 \times m_6$, where $m_1 \times m_2$ is the resolution of the spatial domain in the data point, $m_3 \times m_4$ is the resolution of the angular domain, and the spectral and temporal domain resolutions are shown as m_5 and m_6 , respectively. The size of data points depends on the structure of the data set, and the application at hand. For angular dimension, $m_3 \times m_4$, and spectral dimension, m_5 , usually, the intention is to include all the available information to avoid inconsistencies among different views while rendering the light field. However, for high resolution angular or spectral data sets, the data is divided into smaller patches to exploit the coherency in those domains as well. The coherency of the temporal domain, m_6 , is utilized by including a few consecutive frames of the light field video in the data points. The light field data points can be constructed by a non-overlapping sliding window in the 6D light field video data set. The overlapping sliding window has a negative effect on the compression ratio, but it achieves significant improvements for the compressed sensing applications [14, 194]. The framework proposed in this chapter for sparse representation of light field data is not limited to the dimensionality of the data and can handle n D data points.

5.4 Light Field Compression and Rendering

This section explains the main contributions of this thesis for the compression and rendering of visual data. Specifically, a novel framework is proposed for the compression and real-time rendering of light field and light field video data sets by exploiting the large amounts of correlation that exist within a light field data set. The framework is based on a highly efficient multidimensional dictionary learning approach. Figure 5.3 illustrates the framework in different steps.

The first step of the framework creates small n -D data points, where their size depends on the application. Usually, for light field images, the dimensionality of the data points is 5D, and for light field videos, it is 6D, as explained in Section 5.3. The training data points are used for one-time training of a multidimensional dictionary ensemble (MDE). Each dictionary element is a basis function that represents the light field data point along different dimensions. The main purpose of MDE is to create a high degree of sparsity when data points are represented in the transform

Figure 5.3: Overview of our AMDE compression and real-time rendering framework.

domain of the dictionary. The sparsity is a crucial factor for efficient compression as shown by [139, 195, 196] and also in compressed sensing [171, 197].

The framework includes a novel nonlocal pre-clustering step before training the MDEs that increases the sparsity exhibited in the data set. The pre-clustering algorithm minimizes the reconstruction error and the training time, see Section 5.5.1. The effectiveness of pre-clustering has been explored in computer graphics before; for instance, an iterative variant of K-Means [198] has been utilized for clustering light transport matrices [199] and surface light fields [194].

After pre-clustering the light field data, MDE is trained for each pre-cluster. Combining all MDEs creates an aggregated multidimensional dictionary ensemble or AMDE. The memory footprint of AMDE is negligible, and encoding the data points with it provides fast access to a random location inside each data point, which is useful in a real-time reconstruction algorithm.

Every data set contains high-frequency variations added to the original signal during the acquisition process. To decrease the effect of noise on the compression performance and improve the sparsity, applying a denoising operator before encoding is therefore recommended. An in-depth analysis for motivating this step is explained in Section 5.5.5. The sparse coefficients obtained from encoding using AMDE is further compressed by quantization and entropy coding, see Section 5.5.7.

In the reconstruction and rendering algorithm, see Section 5.5.6, the encoded coefficients are decoded and reconstructed in real-time on GPU, from a specific vantage point, and at a given frame. To increase the angular resolution of the light field data set, the reconstructed viewpoints are interpolated during rendering using auxiliary information such as depth. The following sections provide the details and motivations for each step of the proposed unified framework.

5.4.1 Multidimensional Dictionary Ensemble

A multidimensional dictionary ensemble is designed such that the represented data is sparse in its transform domain. In addition to sparsity, a suitable learning-based representation should have the following properties:

- **Small dictionary size:** One of the objectives of the proposed framework is to transfer the data to a decoder for fast reconstruction. Therefore, the size of the learned dictionary should be small enough to fulfill this purpose. This is unlike previous methods where the constructed dictionary size is relatively large [165, 200, 201].
- **Nonlocal clustering:** Using a smaller number of dictionaries compared to the number of data points to represent the data, leads to nonlocal clustering. The reason is that multiple data points share one dictionary, which results in clustering. Since the training algorithm is based on ℓ_0 norm minimization, the clustering is nonlocal. This feature has shown to promote sparsity [194, 202, 203, 204].
- **Representative power:** The ensemble of dictionaries should be able to represent any unobserved data set of the same dimensionality without the need for retraining.
- **Arbitrary dimensionality:** The framework should not be dependent on the dimensionality of the data set. It should adapt to any form of data such as bidirectional texture function (BTF), Bidirectional reflectance distribution function (BRDF), light field images, light field video. We have left the application of AMDE to these data sets for future work.
- **Local reconstruction:** Fast access to independent elements of each light field data point is required for real-time rendering.

Our proposed framework in *Paper C* satisfies the above criteria, which are elaborated on in what follows.

Similar to other learning-based algorithms for sparse representation of a data set, we also require a *training* set and a *testing* set. The training set consists of data points with similar dimensionality and characteristics of the testing set that we wish to compress. The training set denoted as $\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}$, where $\mathcal{L}^{(i)} \in \mathbb{R}^{m_1 \times m_2 \times \dots \times m_n}$ for n D dimensional data consisting of N_l data points.

Our objective is to train an ensemble of $K \ll N_l$ multidimensional dictionaries, such that each data point $\mathcal{L}^{(i)}$ is representative by only one dictionary:

$$\mathcal{L}^{(i)} = \mathcal{S}^{(i)} \times_1 \mathbf{U}^{(1,k)} \times_2 \cdots \times_n \mathbf{U}^{(n,k)} = \mathcal{S}^{(i)} \bigtimes_{j=1}^n \mathbf{U}^{(j,k)}, \quad (5.3)$$

where $\mathcal{S}^{(i)} \in \mathbb{R}^{m_1 \times m_2 \times \cdots \times m_n}$ is a sparse coefficient tensor, and $\mathbf{U}^{(j,k)} \in \mathbb{R}^{m_j \times m_j}$ is the j -th element of the k -th dictionary corresponding to the dimension $j \in \{1, \dots, n\}$. The symbol $\times_j$ defines the tensor-matrix multiplication along the j -th mode.

Learning an ensemble of n -D orthonormal dictionaries for sparse representation of data can be formulated as:

$$\min_{\mathbf{U}^{(j,k)}, \mathcal{S}^{(i,k)}, \mathbf{M}_{i,k}} \sum_{i=1}^{N_l} \sum_{k=1}^K \mathbf{M}_{i,k} \left\| \mathcal{L}^{(i)} - \mathcal{S}^{(i,k)} \bigtimes_{j=1}^n \mathbf{U}^{(j,k)} \right\|_F^2 \quad (5.4a)$$

subject to

$$\left(\mathbf{U}^{(j,k)} \right)^T \mathbf{U}^{(j,k)} = \mathbf{I}, \quad \forall k = 1, \dots, K, \quad \forall j = 1, \dots, n, \quad (5.4b)$$

$$\left\| \mathcal{S}^{(i,k)} \right\|_0 \leq \tau_l, \quad (5.4c)$$

$$\sum_{k=1}^K \mathbf{M}_{i,k} = 1, \quad \forall i = 1, \dots, N_l, \quad (5.4d)$$

where $\mathcal{S}^{(i,k)}$ are the coefficients of the i -th data point when projected onto the k -th dictionary in the ensemble, K denotes the number of dictionaries in the ensemble, and each dictionary consists of n matrices operating along the data point dimensions. Equation (5.4b) ensures the orthogonality of each dictionary. The sparsity parameter τ_l is defined by the user for the training. $\mathbf{M} \in \mathbb{R}^{N_l \times K}$ is a binary matrix called *membership matrix* that assigns each training data point $\mathcal{L}^{(i)}$ to a dictionary $\{\mathbf{U}^{(1,k)}, \dots, \mathbf{U}^{(n,k)}\}$. Equation (5.4d) guarantees that each data point is only represented by one dictionary and not more. The membership matrix can also be interpreted as a clustering matrix that defines which cluster each data point belongs to. Matching the dimensionality of a dictionary with the data set has shown superior performance and lower representation error when compared to the case when the dictionary dimension is lower than the data dimension, e.g. by vectorizing the high dimensional data [165, 205].

The training process for producing MDE, i.e. $\{\mathbf{U}^{(1,k)}, \dots, \mathbf{U}^{(n,k)}\}_{k=1}^K$ by solving Equation (5.4) is denoted as:

$$\{\mathbf{U}^{(1,k)}, \dots, \mathbf{U}^{(n,k)}\}_{k=1}^K = \mathbf{Train} \left(\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}, K, \tau_l \right). \quad (5.5)$$

This function takes the following as the input arguments: the training set, number of dictionaries in the ensemble, and the training sparsity. The dictionaries are

initialized with random orthonormal matrices and the membership matrix $\mathbf{M}$ is initialized with $1/K$. To solve this function, we iteratively update the dictionary matrices and the coefficient tensor using the following update rules respectively:

$$\mathbf{U}^{(k,j)} = \mathbf{Z}^{(j,k)} \left(\left(\mathbf{Z}^{(j,k)} \right)^T \mathbf{Z}^{(j,k)} \right)^{-1/2}, \quad (5.6)$$

$$\mathcal{S}^{(i,k)} = \mathcal{L}^{(i)} \times_1 \left(\mathbf{U}^{(1,k)} \right)^T \cdots \times_n \left(\mathbf{U}^{(n,k)} \right)^T, \quad (5.7)$$

where

$$\mathbf{Z}^{(j,k)} = \sum_{i=1}^{N_l} \mathbf{M}_{i,k} \mathcal{L}_{[j]}^{(i)} \left(\mathbf{U}^{(n,k)} \otimes \cdots \otimes \mathbf{U}^{(j+1,k)} \otimes \mathbf{U}^{(j-1,k)} \otimes \cdots \otimes \mathbf{U}^{(1,k)} \right) \left(\mathcal{S}_{[j]}^{(i,k)} \right)^T, \quad (5.8)$$

and the membership matrix is updated using:

$$\mathbf{M}_{i,k} = \left(\sum_{b=1}^K e^{\beta(\delta_k^i - \delta_b^i)} \right)^{-1}, \quad (5.9)$$

where

$$\delta_k^i = \left\| \mathcal{L}^{(i)} - \mathcal{S}^{(i,k)} \times_1 \mathbf{U}^{(1,k)} \cdots \times_n \mathbf{U}^{(n,k)} \right\|_F^2. \quad (5.10)$$

To reduce the dimensionality of the data and increase the sparsity, we nullify $(\prod_{j=1}^n m_j) - \tau_l$ elements of $\mathcal{S}^{(i,k)}$ that have the smallest absolute value each time after updating the coefficient tensor, $\mathcal{S}^{(i,k)}$. The temperature parameter, β , in Equation (5.9) is initialized with a small value. For a fixed value of β , the MDE matrices, $\{\mathbf{U}^{(1,k)}, \dots, \mathbf{U}^{(n,k)}\}_{k=1}^K$ are sequentially updated until the changes are minimal. Then the temperature parameter increases and the next updating iteration is applied. MDE converges once the membership matrix is binary or near binary, i.e. satisfying the following criteria:

$$\|\mathbf{M} - \lfloor \mathbf{M} \rfloor\|_2 < \epsilon, \quad (5.11)$$

where ϵ is a small nonnegative user-defined value. The operator $\lfloor \cdot \rfloor$ rounds its argument to the closest integer value. The detailed description of this algorithm can be found in the *Paper C* and its supplementary material.

5.5 Aggregated Multidimensional Dictionary Ensemble

Training a multidimensional dictionary is dependent on the size of the training set, N_l , and the number of dictionaries, K , such that more training data and

Input: $\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}, C, N_r, p, \epsilon$

Result: $\mathbf{c} = \mathbf{Precluster}(\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}, C, N_r, p, \epsilon)$

Init.: Rearrange $\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}$ into $\mathbf{F} \in \mathbb{R}^{N_l \times m_1 m_2 \dots m_n}$

- 1 Compute $\mathbf{F} = \mathbf{A}\mathbf{B}^T$ using CoP with p principal components
 - 2 Apply K-Means to rows of $\mathbf{A}$ with N_r clusters
 - 3 The representative set $\{\bar{\mathcal{L}}^{(i)}\}_{i=1}^{N_r}$ is computed as the centroids of $\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}$ using cluster indices returned by K-Means.
 - 4 Calculate the ensemble $\{\bar{\mathbf{U}}^{(1,c)}, \dots, \bar{\mathbf{U}}^{(n,c)}\}_{c=1}^C = \mathbf{Train}(\{\bar{\mathcal{L}}^{(i)}\}_{i=1}^{N_r}, C, \tau_l, \epsilon)$
 - 5 Compute the pre-clustering membership
 $\mathbf{c} = \mathbf{Test}(\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}, \{\bar{\mathbf{U}}^{(1,c)}, \dots, \bar{\mathbf{U}}^{(n,c)}\}_{c=1}^C, \tau_l, \epsilon)$
-

Algorithm 2: The pre-clustering algorithm implementing $\mathbf{c} = \mathbf{Precluster}(\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}, C, N_r, p, \epsilon)$

dictionaries lead to a more representative dictionary ensemble. Due to the high nonlinearity presented in the data, more training sets and dictionaries often do not lead to over-fitting. However, increasing N_l and K results in higher computational cost, which is undesirable. Therefore, to accelerate the training process and improve the reconstruction quality, a pre-clustering step is added into the framework for a predefined number of training data and dictionaries. In the following, we explain the pre-clustering step that leads to creating AMDE, followed by AMDE pruning and utilizing the resulted dictionary to encode and decode the light field data.

5.5.1 Pre-Clustering

The training set, $\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}$ consisting of N_l data points, is divided into *pre-clusters* using our proposed algorithm that ensures each pre-cluster contains data points that are similar in their sparse representation. As mentioned earlier, MDE trains an ensemble of dictionaries by increasing the sparsity and reducing the representation error where each data point is assigned to only one dictionary. This results in MDE to indirectly perform clustering on the input training data as well. Therefore, if we pre-cluster the data before training the dictionaries, a fewer number of iterations will be required for finding suitable dictionaries for each pre-cluster, making the training stage faster. The clustering effect of MDE on each pre-cluster means that there are less variations of sparsity in each cluster. Therefore, each pre-cluster can be represented with a fewer number of basis or dictionaries, which considerably reduces the computational cost of training and encoding stages.

One clustering algorithm that has been used a lot for dimensionality reduction is K-Means and its variants [198]. However, since the K-Means approach is based on the ℓ_2 norm distance metric rather than ℓ_0 norm that promotes sparsity, it cannot improve the computational time of the training algorithm.

For pre-clustering, in the first step, we perform dimensionality reduction of the training set by decomposing it as $\mathbf{F} = \mathbf{A}\mathbf{B}^T$, where $\mathbf{F} \in \mathbb{R}^{N_l \times m_1 m_2 \dots m_n}$ is a matrix with rows storing the vectorized elements of a data point tensor with size $m_1 \times m_2 \times \dots \times m_n$. $\mathbf{A} \in \mathbb{R}^{N_l \times p}$ is the coefficient matrix, and $\mathbf{B} \in \mathbb{R}^{p \times m_1 m_2 \dots m_n}$ contains the basis vectors in its column, and $p \ll \prod_{j=1}^n m_j$. To calculate the decomposition of $\mathbf{F}$, we employ a factorization method called *Coherence Pursuit* (CoP) [206]. CoP is a fast and simple extension of PCA that is very robust to outliers and noise common in visual data captured with camera sensors.

As the next step in the pre-clustering algorithm, we employ K-Means to find a small subset of the training data points that best depicts the data. We apply K-Means with $N_r \ll N_l$ clusters to group the p -dimensional rows of the coefficient matrix, $\mathbf{A}$, obtained from CoP method. Using the calculated cluster indices, the centroids of the training set $\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}$ are then estimated. The centroids are called *representative set* and are denoted as $\{\bar{\mathcal{L}}^{(i)}\}_{i=1}^{N_r}$.

We train an MDE with C dictionaries on the representative set, where C denotes a user-defined constant for the number of pre-clusters. Afterwards, in the final step, the trained MDE, together with the entire light field data points, are used to create membership vector $c \in \mathbb{R}^{N_l}$ that assigns each data point to a pre-cluster. This step is similar to the encoding process that will be explained in Section 5.5.2. The pre-clustering process is shown as the following function:

$$\mathbf{c} = \mathbf{Precluster}\left(\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}, C, N_r, p, \epsilon\right), \quad (5.12)$$

in Algorithm 2.

The most important parameters during training are the number of pre-clusters, C , and the number of dictionaries, K , as shown in the evaluation of the parameters in *Paper C*. In general, the value of C and K , are proportional to the reconstruction quality, but it comes as a trade-off with computational time. Table 5.1 shows the result for different scenarios: no pre-clustering, pre-clustering with our proposed method, pre-clustering with our proposed method while the number of dictionaries and pre-clusters are doubled, and using K-Means for pre-clustering. Our proposed method reduces computational time significantly while keeping the reconstruction quality almost intact. Increasing K and C results in better PSNR; however, the training time also increases proportionally. The pre-clustering step has a little computational cost compared to the training time.

Input: The training set $\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}$, number of clusters C , number of dictionaries per cluster K , training sparsity τ_l , number of representatives N_r , number of PCA coefficients p , and an error threshold ϵ .

Result: A dictionary ensemble $\{\mathbf{U}^{(1,k)}, \dots, \mathbf{U}^{(n,k)}\}_{k=1}^{CK}$

Init.: Create data points from input training light field videos, Apply pre-clustering on data points

```

1 for  $c = 1 \dots C$  do
2   | Apply MDE on each pre-cluster  $c$ 
3 end
4 Compute aggregated ensemble  $\Psi$  using Equation (5.13)

```

Algorithm 3: The training algorithm that is performed only once, $\mathbf{c} = \text{Train}(\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}, C, K, \tau_l, N_r, p, \epsilon)$

	C	K	PSNR	time (min:sec)	
				pre-cluster	train
No pre-clustering	1	32	65.00	-	62:31
pre-clustering (Algorithm 2)	8	4	64.65	01:24	6:51
pre-clustering, $2C, 2K$ (Algorithm 2)	16	8	66.66	02:28	14:12
K-Means for pre-clustering ([196])	16	8	63.75	00:32	16:27

Table 5.1: The effect of pre-clustering for a training set of 2D data points. For Algorithm 2 we use $N_r = 2048$, $p = 8$, $\tau_l = \tau_t = 8$, and $\epsilon = 10^{-5}$.

Aggregating all MDE dictionaries of all pre-clusters, creates an aggregated multidimensional dictionary ensemble (AMDE) formulated as:

$$\Psi = \bigcup_{c=1}^C \left\{ \mathbf{U}^{(1,k,c)}, \dots, \mathbf{U}^{(n,k,c)} \right\}_{k=1}^K = \left\{ \mathbf{U}^{(1,k)}, \dots, \mathbf{U}^{(n,k)} \right\}_{k=1}^{CK}. \quad (5.13)$$

where, CK is the total number of dictionaries. A summary of the AMDE training stage is given in Algorithm 3. AMDE provides flexibility where we can train the dictionaries only once and re-use it on other data sets with the same dimensionality. One can also combine AMDEs trained on different data sets to create a new set of dictionaries without the need for retraining. The AMDE can also be pruned to represent a data set more efficiently, which is described in Section 5.5.4.

Figure 5.4: Visual quality comparison for Painter [207], a natural light field video data set. See the supplementary video of Paper C for real-time playback of the this data set and Table 5.2 below for quantitative evaluations.

	Uncompressed size: 20604MB				
	size	PSNR	SSIM	t_e	t_d
AMDE	941MB	38.25	0.9286	12436	572
K-SVD	942MB	38.12	0.9281	67235	1465
6D DCT	942MB	36.91	0.9189	489	532
HOSVD	941MB	36.79	0.9147	792	555
CDF 9/7	941MB	31.69	0.8222	974	371

Table 5.2: Results for the natural light field video data set, Painter [207]. Encoding and decoding times are denoted t_e and t_d , respectively, and measured in seconds.

5.5.2 AMDE Encoding

The acquired AMDE from the training step is used to encode the test set denoted as $\{\mathcal{T}^{(i)}\}_{i=1}^{N_t}$, containing N_t data points. The data points are projected onto the AMDE using Equation (5.7) and a dictionary that best represent each data point is chosen. The selected dictionary should lead to the most sparse coefficients with the least reconstruction error. This procedure is represented as the following function:

$$[\{\mathcal{S}^{(i)}\}_{i=1}^{N_t}, \mathbf{m}, \mathbf{z}] = \text{Test}(\{\mathcal{T}^{(i)}\}_{i=1}^{N_t}, \Psi, \tau_t, \epsilon), \quad (5.14)$$

where the inputs to the function are the data points of the testing set, AMDE, and thresholds for sparsity and error, respectively. We set the upper bound of the testing sparsity by τ_t , i.e. the maximum number of nonzero coefficients for each testing data point. Usually the sparsity is lower than this upper bound. The function **Test**(.) outputs a collection of sparse coefficient tensors, $\{\mathcal{S}^{(i)}\}_{i=1}^{N_t}$, and the membership vector, $\mathbf{m} \in \mathbb{R}^{N_t}$, associating the data points with the dictionaries in AMDE. Each element of the membership vector is an index to the best dictionary that describes the corresponding data point. The number of nonzero coefficients in each data point is stored in $\mathbf{z} \in \mathbb{R}^{N_t}$.

5.5.3 AMDE Decoding

To reconstruct a light field after the encoding procedure, we need the set of sparse coefficient tensors, as well as the AMDE. Each data point is estimated as $\hat{\mathcal{T}}^{(i)}$ by simply multiplying the sparse coefficients, $\mathcal{S}^{(i)}$, with its corresponding dictionary, $\mathbf{m}_i$, in the ensemble $\mathbf{U}^{(1,1)}, \dots, \mathbf{U}^{(n,CK)}$ as follows:

$$\hat{\mathcal{T}}^{(i)} = \mathcal{S}^{(i)} \times_1 \mathbf{U}^{(1,\mathbf{m}_i)} \times \dots \times_n \mathbf{U}^{(n,\mathbf{m}_i)}. \quad (5.15)$$

We will explain later how this equation can be modified to reconstruct a single element of the light field video data set in Section 5.5.6.

Figure 5.4 and Table 5.2 present the result of compressing natural light field videos using AMDE framework (without pruning the AMDEs). The result from AMDE is compared with analytically-driven dictionaries such as 6D Discrete Cosine Transform (DCT), and CDF 9/7 wavelets [208] used in standard JPEG2000 image compression standard [209], and learning-based dictionaries such as KSVD [165], and HOSVD. As mentioned earlier, AMDE has a small memory footprint compared to other methods; as an example, the dictionary size for AMDE is 0.0627MB, while the dictionary size for K-SVD is 432MB for the results presented in Table 5.2.

5.5.4 Dictionary Ensemble Pruning

As discussed earlier, increasing the number of dictionaries, i.e., CK , in AMDE can improve the reconstruction quality and compression ratio. However, it also proportionally increases the computational complexity of encoding unobserved data. To solve this problem, a pruning phase is added to the framework to select the most distinct dictionaries within AMDE as the representative dictionaries. To do so, a clustering algorithm is applied on AMDE to group dictionaries that are similar to each other without affecting the reconstruction quality and compression ratio. Intuitively, we would like to choose the most diverse dictionaries in the ensemble that represent the essence of the data set. One such metric for clustering is mutual

Figure 5.5: The light field video data sets used in this section for the evaluation of AMDE framework. From left to right: Painter [207], Chess [211], and Heart (Paper D). Top: single light field view of a frame. Bottom: its corresponding depth image.

<i>Chess</i>	Time	PSNR	Size
AMDE Pruning	140.23s	43.03dB	41MB
AMDE Without Pruning	556.79s	43.01dB	40MB
<i>Heart</i>	Time	PSNR	Size
AMDE Pruning	96.93s	42.93dB	205MB
AMDE Without Pruning	388.06s	42.96dB	182MB

Table 5.3: Ensemble pruning results by a factor of 4 for two data sets: Chess and Heart (see Figure 5.5). The size is calculated as the storage cost of nonzero coefficient values and their corresponding location.

coherence (MC) [210], which defines the maximum cross-correlation between the columns of a matrix $\mathbf{X} \in \mathbb{R}^{n \times n}$ as:

$$\mu(\mathbf{X}) = \max_{1 \leq u \neq v \leq n} |\mathbf{X}_u^T \mathbf{X}_v|, \quad (5.16)$$

where $\mathbf{X}_u$ is the u th column of $\mathbf{X}$. For AMDE, the mutual coherence of each dictionary element is zero as it is an orthogonal matrix. Mutual coherence on a pair of matrices is defined as:

$$\mu(\mathbf{X}, \mathbf{Y}) = \max_{1 \leq u, v \leq n} |\mathbf{X}_u^T \mathbf{Y}_v|. \quad (5.17)$$

To prune the AMDE dictionaries, one can use average mutual coherence (AMC) [212] which is defined as follows:

$$\mu_{\text{avg}}(\mathbf{X}, \mathbf{Y}) = n^{-2} \sum_{u, v=1, u \neq v}^n |\mathbf{X}^T \mathbf{Y}|_{u, v}. \quad (5.18)$$

However, this metric is suitable for pair of matrices while each dictionary in AMDE consist of 6 matrices for a 6D light field video: $\{\mathbf{U}^{(1,k)}, \dots, \mathbf{U}^{(6,k)}\}_{k=1}^K$. Therefore, a pair-wise distance for each dimension j of dictionaries can be computed as following:

$$\mu_{\text{avg}}(\mathbf{U}^{(j,k)}, \mathbf{U}^{(j,k)}) = n_j^{-2} \sum_{u,v=1}^{n_j} \left| \left(\mathbf{U}^{(j,k)} \right)^T \mathbf{U}^{(j,k)} \right|_{u,v} \quad \forall k \in \{1, \dots, K\}. \quad (5.19)$$

To find the most diverse dictionaries based on the metric defined in (5.19), we use K-Means++ [213]. To initialize the cluster centers for K-Means++, dictionaries in AMDE are selected randomly to start the calculation of the pairwise distances. However, we found that random initialization creates fluctuations in the reconstruction quality based on the choice of the first dictionary. To alleviate this problem, the membership matrix, $\mathbf{M}$ defined in (5.4d) is used to select the most frequently used dictionaries in AMDE as the starting point of the K-Means++ algorithm. The result of dictionary pruning is shown in Table 5.3 where the encoding time is improved by a factor of 4 for the data sets *Chess* [211] and *Heart* shown in Figure 5.5.

5.5.5 Denoising Prior to Compression

Light fields contain high-frequency variations that can affect and deteriorate the sparse representation of the data and lower the compression performance. The effect of noise on the efficiency of the sparse representation and the reconstruction quality has been thoroughly studied from a theoretical standpoint [156, 210].

To see the effect of noise on the performance of the training and testing stages of AMDE, we performed a study by applying a denoising operator with different levels of denoising on both the training and test sets. We compare the training and testing performance of the denoised data sets with the original noisy data sets. For choosing a denoising operator, there is a large number of algorithms that can be employed in this study. Our first criteria for selecting a denoiser was finding a method that keeps the main underlying structure of the signal intact while removing the high-frequency noise. Another criterion was the ability to process large data sets, such as light field videos. As a result, a deep learning method called FFDNet [214] was chosen as the representative state-of-the-art denoiser that is suitable for our task.

To evaluate the effect of noise, we carried out the following experiments: setting denoising strength parameter to $\sigma = \{3, 5, 10, 15\}$ for both training and testing set, where $\sigma = 3$ is the lowest, and $\sigma = 15$ is the highest amount of denoising. In the AMDE training algorithm we set the following parameters: 12 frames of light field video, 32 dictionaries with parameters, $C = 4$, $K = 8$, and sparsity $\tau_l = 64$. In the testing procedure, 50 unseen frames of the same data sets were used for evaluation.

Figure 5.6: (a)-(b) The effect of image noise on training. The plots show that the compression efficiency stays relatively flat for all denoising levels on the training set, and that the denoising of the testing data leads to a higher degree of sparsity and better PSNR. For denoising we used FFDNet [214] with the parameter $\sigma = 3, 5, 7, 10, 15$. (c) Shows the effect of image noise on testing (compression) of Painter and Chess data sets shown in Figure 5.5. The plot shows that the PSNR increases along with σ .

The testing sparsity and threshold are set to $\tau_t = 256$ and $\delta_t = 1e - 4$ for *Chess* [211] and $\tau_t = 512$, $\delta_t = 5e - 5$ for *Painter* [207]. The compression efficiency is then measured as PSNR / file size, i.e. taking into account the sparsity achieved by each configuration.

The plots illustrated in Figure 5.6(a)-(b), indicate that denoising the training set has a negligible impact on the compression efficiency of the AMDE algorithm, meaning that the training algorithm is robust to the effect of noise. However, when the evaluation concerning the testing set is performed, we see that the compression efficiency or PSNR / file size substantially increases, as shown in Figure 5.6 (a)-(b) and in Figure 5.6 (c) for the PSNR. More experimental results are shown in

Paper D.

Another interesting observation is that the effect of noise on encoding the testing set reduces significantly if the light field data set is intrinsically sparse such as *Chess* data set. The *Chess* data set has a higher overlap between the neighboring views compared to the *Painter* data set with disparities up to 180 pixels.

5.5.6 Real-Time Decoding and Rendering of Light Field Video

Angular super-resolution of light fields requires the utilization of the depth information in the light field, either by calculating the depth while rendering or using pre-computed or captured depth. The auxiliary information that accompanies light fields, e.g. depth, can be compressed alongside the light field images in a single process. When creating the data points, one can include the depth as a color channel; for instance, the *Painter* data set consisting of depth maps and images, is divided into data points with size $10 \times 10 \times 4 \times 4 \times 4 \times 4$, that includes 10×10 samples of the spatial domain, 4×4 of the angular domain, 4 color channels (depth as the 4th channel), and 4 frames along the temporal domain.

A light field video data set can be displayed in real-time for a virtual point of view, which corresponds to a 2D slice through the data for a given viewing direction and time. As mentioned earlier, AMDE dictionary has a very small memory footprint which can be easily uploaded on the GPU memory together with the sparse coefficients corresponding to frame t of the light field video. Using a pixel buffer object (PBO), the frames are streamed to the GPU for reconstruction and rendering. Once the data is uploaded on the GPU, each i th data point, $\mathcal{T}^{(i)}$, can be reconstructed by a simple multiplication of the coefficients, $\mathcal{S}^{(i)}$, with its corresponding dictionary as explained in Equation (5.15). For a 6D light field video, Equation (5.15) is simplified to:

$$\hat{\mathcal{T}}^{(i)} = \mathcal{S}^{(i)} \bigotimes_{j=1}^6 \mathbf{U}^{(j, \mathbf{m}_i)}. \quad (5.20)$$

This equation reconstructs the entire data point that contains several thousands of elements, which is not suitable for real-time reconstruction and rendering. Recall that each data point contains all the angular information of a light field video frame. However, during rendering, we only require one light field view for a specified frame. Therefore, reconstructing only a fraction of $\mathbf{T}$ is a requirement for real-time rendering. In order to reconstruct only one element of the data point with arbitrary

Figure 5.7: (a) Proxy plane placed in front of the scene, (b) bilinear interpolation coefficients α, β , and (c) view generation by projecting from image coordinates to world coordinates and vice versa, moving point x_A and x_B towards the projection points from the nearby views.

location $x_1, \dots, x_6$, we propose the following formulation:

$$\hat{\mathcal{T}}_{x_1, \dots, x_6}^{(i)} = \sum_{j=1}^{\tau_i} \mathcal{S}_{l_1^j, \dots, l_6^j}^{(i)} \times_1 \mathbf{U}_{x_1, l_1^j}^{(1, \mathbf{m}_i)} \dots \times_6 \mathbf{U}_{x_6, l_6^j}^{(6, \mathbf{m}_i)}, \quad (5.21)$$

where $l_1^j, \dots, l_6^j$ describe the location of j th nonzero element in the sparse tensor $\mathcal{S}^{(i)}$. Equation (5.21) enables fast access to each individual elements of the light field video which is used for fast reconstruction and rendering of each frame.

View Interpolation The angular resolution of the acquired light field images and videos are usually very low. Low angular resolution creates gaps between the sampling views, which results in sudden shifts when the virtual camera moves from

Figure 5.8: The histogram of the nonzero coefficients of a spherical light field data set called TOY (shown on the left) with the angular resolution of 13×360 and spatial resolution of 800×600 .

one light field view to another. Creating novel views for a seamless rendering of light field data sets is an active research topic [53, 215, 216, 217, 218]. However, the majority of these techniques do not allow real-time playback of light field videos. Here, in order to evaluate the performance of our decoding method, we implemented a bilinear interpolation between the four nearest sample points.

We place a proxy geometry in the middle of the scene to guide the rendering process and act as a surface on which the real scene is placed. Any simple geometry such as plane, sphere, cube, or cylinder can be used as the proxy geometry. Since we typically encode the depth map together with the light field data set, we can employ the reconstructed depth map on the GPU to correct the proxy geometry for better forward and backward projection of each point into the real-world and the camera sensors. Please note that the depth map is also reconstructed in real-time on the GPU using Equation (5.21).

The closest cameras to the virtual view are found using the calibration matrices from the sampling points, see Figure 5.7(a). Using the four closest cameras, C_A, C_B, C_C, C_D , and their corresponding distance to the virtual camera, as shown in Figure 5.7, we apply a bilinear interpolation to reconstruct a novel view between the sampled view points using a plane as a proxy geometry:

$$I = (1 - \beta) * (\alpha * I_{C_A}(\hat{x}_{C_A}) + (1 - \alpha) * I_{C_B}(\hat{x}_{C_B})) + \beta * (\alpha * I_{C_C}(\hat{x}_{C_C}) + (1 - \alpha) * I_{C_D}(\hat{x}_{C_D})), \quad (5.22)$$

where parameters, α and β determine the distance of the virtual camera from C_A and C_C , respectively. Moreover, $\hat{x}_A, \hat{x}_B, \hat{x}_C$, and $\hat{x}_D$ are spatial positions of each

Figure 5.9: Spherical light field data set with depth map from left to right: POT, STOOL, VASE, LAMP, and TOY.

	Data set	POT	STOOL	VASE	LAMP	TOY
AMDE	Comp. Ratio	1022:1	58:1	129:1	59:1	148:1
	PSNR(dB)	53.11	39.72	47.23	35.57	44.71
AMDE + Quant. + Huff.	Comp. Ratio	3054:1	130:1	294:1	128:1	329:1
	PSNR(dB)	52.41	39.18	45.96	35.32	44.15

Table 5.4: The comparison of AMDE results with and without quantization and entropy coding of the sparse coefficients using the Huffman coding. The data sets include depth information, see Figure 5.9. All data sets are with spatial resolution of 800×600 except for POT which is 4000×3000 .

camera after the reference point is moved towards the projection points of other cameras.

By sorting the sparse coefficients in descending order before uploading them to the GPU, the user can interactively control the number of coefficients that are employed during the real-time reconstruction. This provides a trade off between reconstruction quality and performance of rendering.

5.5.7 Quantization and Entropy Coding

The sparse coefficients obtained by applying the AMDE encoding algorithm(4.5), can be further compressed through a quantization and entropy coding procedure. Figure 5.8 illustrates the distribution of the nonzero coefficients for the TOY data set (spherical light field, see Section 5.6), where a large number of coefficients are close to zero. This motivates the entropy coding of the coefficients. As the coefficients are in floating points precision, it is required to quantize them before applying any entropy coding algorithm on them.

Figure 5.10: *Quantization of sparse coefficient and its effect on the reconstruction quality for the TOY data set. Top row: the images and their corresponding reconstruction error. Bottom row: The depth maps and their corresponding reconstruction error.*

We employ the Fisher-Jenks classification algorithm [219] that classifies the features of a 1D vector by finding the natural breaks in the data values by minimizing the sum of the squares of the deviations from the class means. The user defines the number of cluster centroids. It should be noted that besides the nonzero coefficients, we also store other information such as the locations of the nonzero elements in each light field data point. After quantization of the nonzero coefficients to 8 bits per coefficient, the Huffman algorithm is applied to the quantized coefficients together with the corresponding locations. The location of a nonzero coefficient is a tuple of integers. For instance, for a 4D light field, the location of a nonzero coefficient consumes e.g. 32-bits, i.e. 8-bits per dimension.

The Huffman coded coefficients are stored on the disk and decoded once they are loaded into the memory. The compression ratio after applying quantization and entropy coding improves about three times while preserving the quality of the reconstruction, as shown in Table 5.4 and Figure 5.10.

5.6 Inward Looking Spherical Light Fields

To capture light fields with larger baselines, one method is to employ many cameras positioned around an object [47] or to use a single camera that rotates around the object [190]. These two methods can also be combined to create a light field acquisition device where multiple cameras are placed on a robotic arm that rotates with a high-precision motor and captures the scene as illustrated in Figure 5.11. We simulated the proposed design using synthetic light fields that were generated

Figure 5.11: *Left: schematic of our 360° inward looking large scale light field capturing system that consists of a precision guided rotating arm with multiple cameras. Right: a few samples of the synthesized light field images from different data sets and their corresponding depth images.*

by placing virtual cameras in Maya and rendering multiple scenes, including objects with various materials. The light fields captured by this method has a very large memory footprint. If 13 cameras are attached to the rotating arm, and the camera movements are one degree apart in the azimuthal direction, then the angular resolution of the captured data is 13×360 . We use AMDE for compression and rendering of these data sets.

Since the angular resolution of these types of light fields is much higher than other data sets, we cannot create data points that include all angular information in it. For this specific data set, the angular resolution along the azimuth is higher than the elevation. Hence, we create data points that include samples for each elevation ring along the azimuth coordinate. We include every 3 – 5 consecutive sampled views on the azimuth coordinate in the data points depending on the structure of the captured scene.

Table 5.4 shows the result for compression of spherical light field data sets using the AMDE framework. The compression ratio, without quantization and entropy coding, varies from 58:1 to 1022:1 depending on the resolution of the data set and its characteristics. The light field is also rendered in real-time with view interpolation using the method described in Section 5.5.6.

5.7 Summary and Future Work

This chapter presented a pipeline for sparse representation and compression of light field data sets. The proposed framework provides efficient and effective compression of a variety of light field data types using AMDE. With GPU implementations, light fields can be displayed in real-time from any vantage point, which can be used for interactive applications such as virtual reality (VR). Novel view generation

is possible due to the fast access to the individual elements of the light field in real-time, which was shown with a bilinear interpolation method.

The novel pruning method improves the computational complexity of the AMDE. We showed that the AMDE is very robust to the effect of camera noise in the training phase, which can be explained with the novel pre-clustering step that increases the sparsity of the representation. On the other hand, reducing the noise in the testing phase can improve the compression ratio significantly.

In the future, it would be interesting to perform novel view generation in the sparse domain of the dictionaries to improve the reconstruction performance. Adding the dictionary similarity as a constraint on the AMDE optimization is another direction for future work. Accurate estimation of the noise parameters for each data set can be used together with the sparsity condition to develop a new sparse recovery algorithm, similar to [156, 220, 221, 222]. The AMDE method performs best when there is considerable overlap between the neighboring views. Therefore, adapting the data points shape to the features of the data set by employing information such as optical flow or disparity map is expected to improve the efficiency of the algorithm.

Compressive Light Field Imaging

Capturing high-quality light fields with a single sensor is a challenging task. Various optical designs have been proposed in the past two decades to record a light field of a scene. One of these designs is *plenoptic camera*, also known as *integral imager* in the optics community. While these optical architectures are capable of sampling the plenoptic function in one snapshot using a single sensor, the resulting measurement is usually a low-dimensional representation with a trade-off between the spatial and angular resolutions. Another design that utilizes the redundancies presented in the light field of a natural scene is called *coded-aperture* photography, where the light field is measured compressively as a two-dimensional projection of light rays using a known mask.

Compressive light field imaging is an attempt to employ an *on-sensor* compression approach to measure just enough samples to be able to reconstruct a full-resolution light field. This chapter explores different design concepts for sampling the plenoptic function and introduces a compressive sensing model for designing a light field video camera.

6.1 Outline and Contributions

The basic concepts of compressed sensing have been explained in detail in Chapter 4. This chapter utilizes the idea of compressive sampling for designing a single sensor light field video camera with a new sensing model. The proposed sensing model presented here is formulated to increase the sparsity and incoherence in the measurements, in order to improve the reconstruction quality. Our novel sensing

model is tested with two camera configurations, where a color-coded mask is placed in front of either a monochrome sensor or a sensor equipped with the CFA. We utilized multiple frames of 2D raw images from either of these two configurations to reconstruct a full resolution light field video. The contributions presented in this chapter are included in *Paper F*.

This chapter is organized as follows: Section 6.2 presents an overview on the light field acquisition methods, Section 6.3 provides details for designing a compressive light field video camera, and finally a summary and possible future extensions of this method are explained in Section 6.4.

6.2 Light Field Acquisition Techniques

The history of light field imaging is more than a hundred years old, where the first attempts were in the area of acquisition. Various methods are proposed, from massive camera array systems to hand-held light field cameras. Capturing high resolution light field and light field videos requires an excessively large amount of bandwidth, which has been the main barrier for light field imaging. Light fields and light field videos have varieties of applications such as refocusing [21], interactive 3D video [223], free-viewpoint television (FTV) [224], light field display [225], auto-stereoscopic 3D display [226], depth estimation and occlusion modeling [227, 228, 229], scene flow estimation [230], light field segmentation [231], stitching [232], material recognition [233], compositing light field video [217], and photo-real digital actors [234].

In the past two decades, there has been an extensive amount of research for developing methods and technologies to capture different dimensions of the plenoptic function. Each method is designed to sample the light fields based on different parameterizations. Light fields can be sampled uniformly or non-uniformly (unstructured). In this thesis, we consider only uniform sampling.

Designing a light field acquisition system depends on a few important factors, such as the baseline between the neighboring views, spatial and angular resolutions, and the ability to capture a light field video. These factors make the capturing design suitable for a specific application. As an example capturing wide-baseline is suitable for applications such as 3D Television (3DTV), Free-view Television (FTV) [224], and augmented reality (AR) applications. The ideal capturing device would sample densely over all viewpoints. However, the large volume of data and high capturing costs are the main limitations of this process. Light field capturing systems can be classified into two main categories as single-sensor and multi-sensor systems, which are explained in the following.

Figure 6.1: Multi sensor light field camera examples. (a) *Fuji 100 cameras* [235], (b) *Stanford camera array* [47], (c) *Linköping University light stage device suitable for large scale objects*, (d) *Light field acquisition for heart surgery (Paper D)*, (e) *Pelican camera*.

6.2.1 Multi Sensor LF Acquisition

Light field acquisition with multi-sensor design was first developed by Wilburn et al. [47], see Figure 6.1 (b), and Zhang et al. [236]. Capturing concentric mosaic with an array of cameras placed in front of the scene in a circular or planar arrangement was introduced long before, and was used in movies such as *Matrix* [51]. Marton et al. [237] explored a multi-viewing system for real-time capturing and reconstruction of light fields and displaying it with full horizontal parallax on a multi-projector light field display. Camera arrays have been used on mobile phones as well, such as Pelican [238], where the phone is equipped with an array of small cameras, as shown in Figure 6.1(e). Multi-camera system has been used to create life-size digital humans by capturing real people in a light stage environment and displaying them as a hologram using an automultiscopic projector array [239]. Recently, Overbeck

et al. [240] presented a system for acquiring, processing, and rendering panoramic light field stills. Their capturing system consisted of two designs with multiple cameras for capturing spherical light field images.

The design introduced in Section 5.6 makes it possible to capture spherical light fields of large scale objects and artifacts. A prototype of this design is shown in Figure 6.1 (c), where the cameras are attached on a motorized circular arm. Multi-camera systems have also been used for capturing light field video of heart surgery, as shown in Figure 6.1(d) and presented as an application in *Paper D*.

Multi-sensor designs are capable of recording a dynamic scene in high spatial resolution and create contents that provide plausible and immersive experiences [235, 241]. These systems provide a very large baseline in the captured data, which increases the accuracy of the estimated depth layers, and is suitable for recovering the 3D structure and dense 3D motions of the dynamic nonrigid scenes [242]. However, the disadvantage of these systems is that they are often very bulky and expensive and require careful calibration and maintenance, see Figure 6.1(a) as an example where 100 cameras are arranged to record the scene.

6.2.2 Single Sensor LFs

Light field capturing with a single sensor is dependent on the optical elements and hardware that are used in designing the capturing system. A few examples of these additional components include micro-lens arrays, attenuation mask, and a robotic arm. In what follows, a few examples of these designs are explained.

Lenslet cameras Lippman was the first to introduce the idea of placing micro-lenses array (MLA) in front of a sensor in 1908 [43], as shown in Figure 6.2 (a). Adelson and Wang [243] proposed a similar and improved design called *plenoptic camera*, where an MLA was placed on the focal point between the objective lens and the sensor, enabling hand-held light field capture. This design was later modified and implemented by Ng et al. [46], leading to the commercial production of *Lytro* and *Raytrix* cameras. Utilizing an array of lens-prism pairs in front of the main lens can turn a conventional high-resolution camera into a light field camera [244]. However, the lens-prisms must be well-corrected for accurate capturing. Micro-lenses have also been used in light field microscopy, where it allows for oblique views and 3D reconstruction and refocusing [10]. A few of these cameras are shown in Figure 6.2(b)-(e).

Since the sensor size is limited, lenslet design suffers from a trade-off between angular and spatial resolution [244]. Another problem associated with lenslet cameras is the narrow baseline where the distance between the neighboring views is minimum, leading to sub-pixel disparities, which reduces the accuracy of the

Figure 6.2: (a) Lippman's 1908 Integral Imaging. Spatially-multiplexed light field capture using lenslet (Fly's Eye lenslet). (b)-(e) Single sensor light field camera with micro-lens array (MLA) design.

depth layers. This design is mainly suitable for recovering a shallow depth map and post-capture focusing.

Coded Aperture Cameras In coded-aperture photography, a known mask is placed between the sensor and the aperture plane. Coded-aperture imaging has been used in many fields such as computational photography [22, 23, 179, 180, 245], astronomy, and medicine [246, 247, 248].

For light field imaging, Veeraraghavan et al. [249] utilized frequency domain and a heterodyne method to derive a simple attenuating mask for designing a light field camera. Higher reconstruction quality can be achieved by using *Liquid Crystal Array* (LCA) to change the mask pattern to capture multiple shots of the scene and increase the number of light field measurements [250]. The spatial resolution of the light field image can improve by employing two masks in the light field camera [251] at the cost of reducing the light efficiency due to the masks.

Another coded-aperture design was proposed by Babacan et al. [252], where a randomly generated mask was placed at the aperture plane, and the final light field image was reconstructed using a Bayesian framework. A compressive sensing framework for capturing light fields as a 2D projection on the sensor was proposed by Marwah et al. [22] where they built a prototype camera using their proposed design. This design was improved later by Miandji et al. [23] by suggesting to use a color-coded mask instead of a monochrome mask and training 5D dictionaries

Figure 6.3: *Single sensor light field gantry. (a) Stanford spherical gantry (b) Lego Mindstorms gantry (c) Incident light field capturing.*

as priors for the reconstruction of the light field image. In contrast to lenslet cameras that suffer from spatio-angular trade-off, coded aperture cameras enable higher spatio-angular resolutions at the cost of higher computational complexity. A coded-aperture solution for light field video acquisition is explained in Section 6.3, where a color-coded mask is placed in front of the sensor. By exploring the temporal correlations between the light field frames, a desirable sparse representative model is developed for recovering the light field frames.

Light Field Gantry Another way to capture light fields is by using a single camera that is pinned to a motorized device that moves in a specified path and captures the scene. Since the camera can only capture one instance at a time, this design is only suitable for static scenes and cannot record dynamic scenes. Levoy et al. [189] attached a camera on a robotic arm to rotate it around a small object to acquire images from different angles, see Figure 6.3 (a). Unger et al. [253] introduced a system for capturing incident light field as a volume by moving the camera and acquiring the spatially varying illumination, see Figure 6.3 (c). Other gantry systems include Lego Mindstorms Gantry¹, Figure 6.3 (b), and marker-less capture [52]. An interactive capturing system that guides the user for sampling light fields using mobile phones was introduced by Mildenhall et al. [216]. They used multi-plane images for synthesizing the intermediate views.

6.3 Compressive Light Field Video Camera

A 6D light field video described with two-plane parameterization is denoted as $\mathcal{L}(s, t, u, v, \lambda, f)$, where (s, t) is the spatial domain, (u, v) describes the angular

¹ <http://andrew.adams.pub/>

Figure 6.4: (a) Sketch of light field camera design with color-coded mask. The mask is placed at distance d_m between the sensor and the aperture plane. Raw images with a color-coded mask acquired with (b) a sensor equipped with CFA and (c) a monochrome sensor.

domain, λ spectral dimension, and f is the time domain. A 2D image is captured by integrating light rays over the angular domain and projecting onto the camera sensor:

$$\mathbf{y}(s, t, \lambda) = \int_{u, v} \mathcal{L}(s, t, u, v, \lambda) \cos^4 \alpha \, du dv, \quad (6.1)$$

where α is the angle between incoming light ray and the sensor, and $\cos^4 \alpha$ is the vignetting effect [254], which we ignore in our model for simplicity. A monochrome coded-mask, Φ , at a small distance, d_m , in front of the sensor, modulates the light rays and projects it onto the sensor [22]. This design changes the integral photography Equation (6.1) into the following:

$$\mathbf{y}(s, t, \lambda) = \int_{u, v} \mathcal{L}(s, t, u, v, \lambda) a(s + \sigma(u - s), t + \sigma(v - t)) \, du dv, \quad (6.2)$$

where the function $a(\cdot)$ is the attenuation mask and $\sigma = d_m/d_a$ defines the shear of the mask pattern. This equation can be written in discrete form as a matrix-vector multiplication:

$$\mathbf{y} = \begin{bmatrix} \Phi^1 & \Phi^2 & \dots & \Phi^\nu \end{bmatrix} \begin{bmatrix} \mathbf{x}^1 \\ \mathbf{x}^2 \\ \vdots \\ \mathbf{x}^\nu \end{bmatrix}, \quad (6.3)$$

where $\nu = |u||v|$ denotes the angular resolution and $\mathbf{x}^i \in \mathbb{R}^\omega$ contains the vectorized light field view images, with $\omega = |s||t|$ as the spatial resolution. The mask model in Equation (6.2), is represented as $\Phi = \{\Phi^i\}_{i=1}^\nu$, with diagonal matrices $\Phi^i \in \mathbb{R}^{\omega \times \omega}$. A sensing matrix corresponding to a monochrome mask should be applied on each color channel independently, which is shown to increase the coherency of the measurements, and in turn leading to a lower reconstruction quality [23]. In contrast, a sensing matrix for a color-coded mask is applied to all color channels of a light field simultaneously.

6.3.1 Sensing Matrix Design

For designing a sensing matrix that is suitable for a compressive light field video camera, we considered two camera configurations: First, a sensor, equipped with a color filter array (CFA) and a color-coded mask that is placed between the aperture and the sensor at a predefined distance d_m as illustrated in Figure 6.4 (a). Second, a monochrome sensor with a color-coded mask placed similarly as the first configuration. The raw input of each configuration is shown in Figure 6.4 (b) and (c). In what follows, we will explain the sensing matrix for each configuration. These matrices are used to reconstruct a full resolution light field video.

Sensor with CFA and color mask Placing a random color mask at a small distance in front of the sensor of a camera with overlaid CFA for compressive imaging was first introduced by Miandji et al. [23]. Their method assumes that the scene is static or nearly static in order to capture multiple snapshots, and thereby increasing the incoherency of the measurements, which leads to the reconstruction of a high-quality light field image. However, this method is not suitable for recording the light field video of a dynamic scene, as capturing multiple shots for each frame is not feasible. The consecutive frames of a light field video, however, contain significant correlations that can be used in the reconstruction algorithm, which is explained in Section 6.3.3. The mask can be changed using a piezo motor to create a unique random mask pattern for each light field frame. The randomness of the mask also increases the incoherency of the measured light field video, which is suitable for compressive light field recovery.

The color-coded mask may contain any wavelength in the spectrum, however, without loss of generality, we assume Red, Green, Blue (RGB) as color channels to simplify the notations. The sensing matrix for each frame $j \in \{1, \dots, N\}$ of a light field video with N frames, using a color-coded mask and a CFA sensor is similar to the definition of Miandji et al. [23]:

$$\Psi^j = \begin{bmatrix} \Phi^{1,R,j} & \dots & \Phi^{\nu,R,j} & 0 & 0 \\ 0 & & \Phi^{1,G,j} & \dots & \Phi^{\nu,G,j} & 0 \\ 0 & & 0 & & \Phi^{1,B,j} & \dots & \Phi^{\nu,B,j} \end{bmatrix}. \quad (6.4)$$

For light field video, we propose to stack vertically β consecutive frame measurement matrices, denoted as Ψ^j , $i \in \{1, \dots, \beta\}$, as follows:

$$\Phi_{(\text{I})} = \begin{bmatrix} \Psi^1 \\ \vdots \\ \Psi^\beta \end{bmatrix} \in \mathbb{R}^{\beta\omega\lambda \times \omega\nu\lambda}. \quad (6.5)$$

The sensing matrices can alternatively be stacked horizontally as:

$$\Phi_{(\text{II})} = [\Psi^1 \dots \Psi^\beta] \in \mathbb{R}^{\omega\lambda \times \beta\omega\nu\lambda}. \quad (6.6)$$

We will see in the next section how horizontal arrangement reduces incoherency of the measurements. This arrangement is equivalent to linear combination of the input frames with a convolution filter over their angular domain into $\mathbf{y} \in \mathbb{R}^{\omega\lambda}$. On the other hand, the sensing matrix $\Phi_{(\text{I})}$ (6.5) contains β times more samples, and as it will be confirmed later, is superior to $\Phi_{(\text{II})}$ (6.6).

Monochrome sensor with color mask Another design for capturing compressive light field video, similar to [179, 255], is to place a color-coded mask in front of a monochrome sensor. In this design, the spectral domain is compressed during capturing, leading to a higher compression ratio when compared to the design with a CFA sensor. This feature is suitable for the fast transmission of the captured data, and in practice, it is easier to place the mask in the desired location without the CFA on the way. Another benefit of the monochrome sensor design is that the light efficiency is higher when compared to CFA equipped sensor.

The sensing matrix for a single frame of a monochrome sensor with an RGB color-coding mask is:

$$\Lambda^j = [\Phi^{1,R,j} \dots \Phi^{\nu,R,j} \quad \Phi^{1,G,j} \dots \Phi^{\nu,G,j} \quad \Phi^{1,B,j} \dots \Phi^{\nu,B,j}]. \quad (6.7)$$

The concatenation of the sensing matrices vertically, $\Phi_{(\text{III})}$, and horizontally, $\Phi_{(\text{IV})}$, is defined as follows:

$$\Phi_{(\text{III})} = \begin{bmatrix} \Lambda^1 \\ \vdots \\ \Lambda^\beta \end{bmatrix} \in \mathbb{R}^{\beta\omega \times \omega\nu\lambda}. \quad (6.8)$$

$$\Phi_{(\text{IV})} = [\Lambda^1 \dots \Lambda^\beta] \in \mathbb{R}^{\omega \times \beta\omega\nu\lambda}, \quad (6.9)$$

It should be noted that these sensing matrices does not require custom hardware implementations and are formulated after the capturing process and only affect the reconstruction algorithm. We assume that only the encoded frames captured with either of the design configurations and their corresponding masks are available to the reconstruction method explained in Section 6.3.3.

6.3.2 Dictionary Training

The inverse problem of light field reconstruction from compressive measurements, Equation (6.3), requires a proper basis function or dictionary $\mathbf{D}$ to represent the natural light field data sparsely. The proposed approach for compressive light field video acquisition trains a dictionary that admits the sparse representation of the data while including the temporal domain in the atoms of the dictionary. Doing so will increase the sparsity due to the correlation between the neighboring frames.

Assuming we have a training set $\mathbf{Z} = \{\mathbf{z}^1 \dots \mathbf{z}^t\}$ consisting of t light field video frames, the dictionary $\mathbf{D}$ is trained using an online dictionary learning algorithm [143] by solving:

$$\min_{\mathbf{D}} \frac{1}{t} \sum_{i=1}^t \min_{\mathbf{h}^i} \frac{1}{2} \|\mathbf{z}^i - \mathbf{D}\mathbf{h}^i\|_2^2 + \lambda \|\mathbf{h}^i\|_0, \quad (6.10)$$

where $\mathbf{h}^i$ is the latent representation for each training data $\mathbf{z}^i$. The non-negative coefficient λ in (6.10) defines a trade-off between the reconstruction error and sparsity.

To solve Equation (6.10), we create small data points from the light field video data set. The dimensionality of the data points defines the quality of the dictionary and how well it represents the data. Marwah et al. [22] show that using 4D spatio-angular data points increases angular coherency in the reconstructed light field. Including the color information in the data points and expanding them to 5D has shown to improve the reconstruction quality even further compared to 4D patches [23]. We propose to include the temporal dimension and expand the data points to 6D such that it spans spatial, angular, spectral, and temporal domains concurrently. Adding temporal information is expected to increase the sparsity, improve the reconstruction quality, and increase the temporal coherency of the reconstructed light field video. The dimensionality of the data points is then denoted as $n = s \times t \times u \times v \times \lambda \times \beta$ corresponding to the spatial, angular, spectral, and temporal resolution of the data point, respectively.

We considered two options for training the dictionaries: *Single-frame* dictionary where the dictionary is trained on 5D data points that are extracted from each individual frame. The atoms of this dictionary is a basis function representing spatial, angular and spectral domains. The second option is to extract 6D data points as explained above to include β -consecutive frames and train a *multi-frame* dictionary with the following structure:

$$\mathbf{D} = \begin{bmatrix} \mathbf{D}_1 \\ \mathbf{D}_2 \\ \vdots \\ \mathbf{D}_\beta \end{bmatrix}, \quad (6.11)$$

where $\mathbf{D} \in \mathbb{R}^{\beta\lambda\nu\omega \times \rho\beta\lambda\nu\omega}$ and ρ is the over-completeness factor. The multi-frame dictionary contains the temporal information which improves the sparsity when compared to the single-frame dictionary.

6.3.3 Sparse Reconstruction

To recover a light field video from the sparse measurements, the reconstruction algorithm or the sensing model needs to take into account the sensing matrices

Figure 6.5: Comparison of proposed sensing models of the Boxer data set [211] on a monochrome sensor with color-coded mask.

Models	<i>Boxer</i>		<i>Chess</i>	
	SSIM	PSNR(dB)	SSIM	PSNR(dB)
Miandji et al. [23]	0.8426	27.31	0.8832	28.75
SM1	0.9023	29.85	0.9201	30.69
SM2	0.8135	25.82	0.8620	27.28
SM3	0.9500	33.18	0.9619	34.49

Table 6.1: Comparison of the proposed sensing models; data sets used are *Boxer* and *Chess* [211] with non-overlapping patches, each with 5 frames for testing. Moreover, we set $\beta = 3$, $n = 7 \times 7 \times 5 \times 5 \times 3 \times 3$, image size: $[540, 960]$, and we used 10 frames for training (distinct from the testing set).

described in Section 6.3.1 and the dictionaries described in Section 6.3.2. As explained in Chapter 4, a signal is recovered from sparse measurements by solving Equation (4.11). For the sake of clarity we revisit the equation here:

$$\arg \min_{\theta} \|\theta\|_0 \quad \text{s.t.} \quad \|\mathbf{y} - \Phi \mathbf{D} \theta\|_2^2 \leq \epsilon, \quad (6.12)$$

We considered three possible sensing models for the reconstruction of the full resolution light field video, and each model will be explained in detail below. These sensing models are applicable to both camera designs that were explained above. However, for simplicity, we only explain the sensing model for a monochrome sensor.

Sensing Model 1 (SM1) In the first sensing model, we append vertically the sensed 2D raw image of each frame, $\mathbf{y}^i$, into a vector $\mathbf{y} \in \mathbb{R}^{\beta\omega}$ and their corresponding

sensing matrices Λ^i are also stacked vertically into matrix $\Phi_{(\text{III})} \in \mathbb{R}^{\beta\omega \times \omega\nu\lambda}$ as explained in Equation (6.8). We train a single-frame dictionary for this sensing model on 5D training data points, to create a dictionary $\mathbf{D} \in \mathbb{R}^{\omega\nu\lambda \times \rho\omega\nu\lambda}$. Utilizing β -consecutive light field frames $\{\mathbf{x}_1, \dots, \mathbf{x}_\beta\}$ in the reconstruction, where $\mathbf{x}_i \in \mathbb{R}^{\omega\nu\lambda}$, leads to formulating the following minimization problem:

$$\arg \min_{\theta} \|\theta\|_0 \text{ s.t. } \left\| \begin{bmatrix} \Lambda^1 \mathbf{x}^1 \\ \vdots \\ \Lambda^\beta \mathbf{x}^\beta \end{bmatrix} - \begin{bmatrix} \Lambda^1 \\ \vdots \\ \Lambda^\beta \end{bmatrix} \mathbf{D} \theta \right\|_2^2 \leq \epsilon, \quad (6.13)$$

where ϵ is the error of the reconstruction that is defined by the user. Arranging the matrices in the vertical form increases the number of incoherent samples which leads to improvement of the reconstruction quality. The result of using *SM1* is illustrated in Figure 6.5, where as it is visible, the sensing model performs quite well in the recovery of the stationary objects in the background but it reconstructs the moving objects with artifacts along the edges due to the lack of temporal information in the single-frame dictionary.

Sensing Model 2 (SM2) To include the temporal information in our sensing model, we train a multi-frame dictionary on 6D data points that spans the temporal domain in *SM2*. However, unlike *SM1*, in this model the elements of the sensing matrix are organized horizontally as Equation (6.9). Therefore the reconstruction depends on solving the following problem:

$$\arg \min_{\theta} \|\theta\|_0 \text{ s.t. } \left\| \begin{bmatrix} \Lambda^1 \dots \Lambda^\beta \end{bmatrix} \begin{bmatrix} \mathbf{x}^1 \\ \vdots \\ \mathbf{x}^\beta \end{bmatrix} - \begin{bmatrix} \Lambda^1 \dots \Lambda^\beta \end{bmatrix} \begin{bmatrix} \mathbf{D}_1 \\ \vdots \\ \mathbf{D}_\beta \end{bmatrix} \theta \right\|_2^2 \leq \epsilon. \quad (6.14)$$

The multi-frame dictionary that is trained on β consecutive frames enables *SM2* to reconstruct the light field frames with less temporal artifacts compared to *SM1*. The quantitative and visual comparison of the two models can be seen in Figure 6.5 and Table 6.1. As expected, horizontal arrangement of the sensing matrix decreases the number of incoherent samples that are used in solving the BPDN problem (6.12). The optimization problem cannot find the suitable coefficients for the reconstruction despite the fact that the dictionary atoms encode multi-dimensional information. As an example, in Figure 6.5, the light field frame is slightly blurry and the colors are not recovered correctly.

Sensing Model 3 (SM3) The advantages and disadvantages presented in *SM1* and *SM2* led us to design a third sensing model to maximize the incoherency of the measurements as well as the sparsity. Therefore, we combine the sensing matrix of *SM1* to have β times more incoherent samples, with the multi-frame dictionary of

Figure 6.6: A window of size $\beta = 3$ for reconstruction is chosen so that the current frame is placed at the center of the window (except for corner cases). For each frame of the original light field video (frame 2 in this example), we reconstruct three light field sequences (three rows shown with dashed lines). Therefore, we can combine three reconstructed frames (shown with a red box) to obtain a single frame corresponding to frame 2 in the original light field video.

SM2 to benefit from the temporal correlations in our optimization problem. Since each light field video frame is captured independently from the other frames, we re-arrange the matrix multiplication of the sensing matrix and the dictionary to achieve the following minimization problem:

$$\arg \min_{\theta} \|\theta\|_0 \text{ s.t. } \left\| \begin{bmatrix} \Lambda^1 \mathbf{x}^1 \\ \vdots \\ \Lambda^\beta \mathbf{x}^\beta \end{bmatrix} - \begin{bmatrix} \Lambda^1 \mathbf{D}_1 \\ \vdots \\ \Lambda^\beta \mathbf{D}_\beta \end{bmatrix} \theta \right\|_2 \leq \epsilon, \quad (6.15)$$

where $\mathbf{D}_i \in \mathbb{R}^{\lambda\nu\omega \times \rho\beta\lambda\nu\omega}$, $i \in \{1, \dots, \beta\}$, are sub-matrices of the multi-frame dictionary $\mathbf{D} \in \mathbb{R}^{\beta\lambda\nu\omega \times \rho\beta\lambda\nu\omega}$, defined in (6.11), corresponding to frame i of the captured light field video. The recovered signal from solving the optimization problem of *SM3* is $\hat{\mathbf{x}} \in \mathbb{R}^{\beta\omega\nu\lambda}$, which means that for each frame of the light field, we recover β frames. By applying a simple averaging method over the β recovered light field frames, we further improve the reconstruction quality, see Figure 6.6 for an illustration of temporal averaging of the estimated light field frames. The result is expected to be improved with a more sophisticated motion-aware algorithm, such as [256]. The superiority of the *SM3* over other sensing models is confirmed in Figure 6.5 and Table 6.1.

To solve the optimization problem presented in the three sensing models, i.e. (6.13), (6.14), and (6.15), we applied the Smoothed- ℓ_0 (SL0) algorithm [160] as it has shown a better trade-off between the reconstruction quality and the speed compared to other sparse recovery methods such as [150, 153, 257, 258, 259].

To evaluate the performance of *SM3*, we simulated the compressive capturing of both sensing configurations of a monochrome sensor and CFA equipped sensor with a color-coded mask. Three light field video data sets were used for this purpose: *Boxer-Gladiator-Irish*, *Chess*, and *Chess-moving* from [211]. We compared the result of reconstruction using *SM3* with temporal window size of 3 ($\beta = 3$), with respect to the method of Marwah et al. [22], Miandji et al. [23], and the deep

Monochrome Sensor						
Data Set	Boxer		Chess		Chess-moving	
Algorithm	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
Marwah et al.	0.4909	21.19dB	0.3954	19.87dB	0.2421	19.27dB
Miandji et al.	0.8426	27.31dB	0.8832	28.76dB	0.9388	31.25dB
Ours (SM3)	0.9500	33.07dB	0.9619	34.34dB	0.9693	34.55dB
CFA Sensor						
Data Set	Boxer		Chess		Chess-moving	
Algorithm	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
Marwah et al.	0.9265	30.74dB	0.9443	32.15dB	0.9722	35.28dB
Miandji et al.	0.9504	34.28dB	0.9627	35.75dB	0.9817	38.15dB
Inagaki et al.	0.9608	33.08dB	0.9683	33.82dB	0.9843	37.53dB
Ours (SM3)	0.9824	40.29dB	0.9860	41.23dB	0.9863	39.91dB

Table 6.2: Dictionary type comparison; **Boxer** and **Chess** data sets; non-overlapping data points with size: $n = s \times t \times u \times v \times \lambda \times \beta = 7 \times 7 \times 5 \times 5 \times 3 \times 3$, number of frames for testing: 5, number of frames for training: 10, image size = [540, 960], reconstruction time is for all frames. The comparison is with Marwah et al. [22], Miandji et al. [23], and Inagaki et al. [176].

learning method of Inagaki et al. [176]. The dictionary D is constructed by training on a set consisting of 10 frames of the *Boxer-Gladiator-Irish*, and 10 frames of *Chess*. The evaluation is done on unseen frames from all three data sets. The result is summarized in Table 6.2. The reconstruction quality of the monochrome sensor compared to the CFA equipped sensor is lower as expected since the CFA sensor contains three times more samples. For more details about our algorithm and the results, see *Paper F*.

The effect of temporal window size is shown in Figure 6.7. Although temporal window size, β , is context-dependent, the reconstruction quality improves by including more samples from the neighboring frames. However, this comes at the cost of a larger dictionary and higher computational complexity.

6.4 Summary and Future Work

This chapter presented a compressive solution for designing a single sensor light field video camera. Similar to previous compressive coded-aperture imaging, a random color mask is placed between the sensor and the aperture plane. To enforce the redundancy to the measured data for recovering the light field video frames, we stack a few consecutive measured frames to create a sensing model that explores the temporal correlations of the video. Since the mask is random for each captured frame, stacking the sensing matrix and measurement also improves the incoherency

Figure 6.7: Temporal window size affects the reconstruction quality for both configurations of color-coded mask with monochrome sensor and sensor equipped with CFA. The result is for Boxer-Gladiator-Irish data set [211].

of the measurements. By training a multidimensional dictionary that includes temporal information in its atoms, we were able to reconstruct a full resolution light field video with minimum artifacts. The result confirms that the proposed sensing model can handle movements in the scene, even in challenging scenarios, when the camera moves rapidly within the scene.

Since the representativeness of the trained dictionary is dependent on the motions in the scene, incorporating a motion aware clustering (i.e. disparity or optical flow) into the training and reconstruction step can benefit the algorithm. Interpolation between the reconstructed frame using optical flow is also another interesting research direction. As we showed in Figure 6.7, the temporal window size affects the quality of reconstruction, and it is highly dependent on the motion in the scene. Therefore, adapting the window size to the context of the frames is a natural direction for future research.

Concluding Remarks and Outlook

This dissertation provided algorithms that improve computational photography techniques for high dynamic range imaging and light field photography. This final chapter gives a summary of the research contributions of this thesis and presents discussions for future research directions.

7.1 Summary of Contributions

We discussed the challenges for accurate imaging of natural scenes using conventional cameras. Specifically, we identified problems for reconstructing a high dynamic range image from a spatially multiplexed image with multiple exposures. For light-field imaging, we described the current obstacles and provided solutions for capturing, compression, and rendering of a variety of light field data sets. We summarize the contributions of this thesis below:

- We showed that capturing high dynamic range images with a single sensor is feasible using an accurate and robust noise-aware HDR reconstruction framework that performs operations such as re-sampling, color interpolation, denoising, and HDR reconstruction in a single unified kernel function that adapts to the features of the image.
- Related to light-field imaging, we provided a solution for the challenging task of acquiring light field videos due to the excessive bandwidth requirements of such data. A robust compressive sensing model was introduced for two

camera designs with monochrome sensors and sensors equipped with CFA, where an attenuation color-coded mask is placed between the aperture plane of the camera and the sensor, with a user-defined distance from the sensor.

- We demonstrated how our proposed compression framework using an ensemble of multidimensional dictionaries outperforms prior work for compressing high resolution light field and light field video data sets. The presented framework provides random access to the memory, which makes it suitable for real-time reconstruction, novel view synthesis, and rendering.
- Finally, we presented various applications of our compression method from large scale 360° inward-looking light field acquisition devices to light fields captured with hand-held and multi-camera systems.

7.2 Future Work

The computational methods introduced in this dissertation provide a guideline for future research. The field of HDR imaging has matured during the last decade, leading to commercial realization of HDR televisions, monitors, and cameras. Despite these advancements, there are still many challenges for capturing high dynamic range videos on conventional cameras. There have been a few attempts to acquire HDR images of dynamic scenes such as [260, 261], but these solutions are not suitable for capturing HDR video on consumer cameras. One possible research direction to solve this challenge is to employ the spatially multiplexed images in a sequence with various exposure settings on a single sensor, e.g. by utilizing the dual-ISO technique, and reconstruct the interlaced frames using our unified framework for HDR reconstruction with three-dimensional kernels including both spatial and temporal information. The sparse representation can also be utilized for the reconstruction of HDR video from dual-ISO frames, similar to [262]. Another interesting research direction for the HDR reconstruction method is to explore the similarity of nonlocal regions of the image and incorporate that in the reconstruction algorithm. This is similar to modern denoising algorithms, and might also solve the problem with regions where both exposures are saturated.

The compressive light field video camera can potentially be extended for capturing higher dynamic range light field videos. One approach can be to modify the compressive sensing model to take into account the interlaced multi-exposure frames, similar to the dual-ISO video, in both capturing and reconstruction steps. We explored the proposed sensing model for two camera designs, where a color-coded mask is placed in front of the sensor of both configurations. An interesting extension of this method is to place a random color-coded mask [61] in front of the sensor to act as the CFA. More interestingly, one can optimize this random CFA

Figure 7.1: Total number of film, DSLR, mirrorless and smartphone cameras sold globally. Source: CIPA¹

over a large data set of light fields to improve the incoherence and reconstruction quality. Indeed we can also move the CFA using a piezo motor for enhanced incoherence in the measurements. We showed in Chapter 6, that the temporal window size affects the quality of reconstruction. However, that also depends on the scale of motion in the scene. Therefore, it would be natural to adapt the temporal window size to the movements in the scene (or the camera) by considering the optical flow into our model for training a dictionary and reconstructing the frames.

Light field data sets encode a large amount of hidden information that can be explored to derive explicit representations such as depth map, optical flow, and BRDF. There have been a few articles on estimating SVBRDF from light field data sets [55]. This can be a very interesting future research direction.

As computational photography is mainly intended for our visual system, visual perception must be considered when designing a computational framework. As an example, we are utilizing a denoiser before applying AMDE encoding on the light field data set. The denoising can be applied based on the human visual system such that noise with visual significance in the data are removed, similar to [263].

The proposed compression framework, AMDE, can be further improved by enforcing the dictionary similarity metric in the training step and optimizing the number of dictionaries in the AMDE.

Since the introduction of mobile cameras in 2008, there has been a massive surge

¹ <http://www.cipa.jp/stats/documents/common/cr1000.pdf>

Figure 7.2: *There has been a huge surge worldwide in the number of smartphones with dual, triple, and multi cameras. Note the significant increase in the number of dual camera smartphones since 2016. Source: Statistia © 2019*²

in mobile photography, as shown in Figure 7.1. This has led to a decrease in the sales of professional digital cameras. The landscape of imaging is shifting towards mobile photography, which demands to be in the center of attention in the future research directions. Enabling onboard processing is a necessity for modern mobile phones, where the user will be able to capture professional images with minimum effort. Currently, there is a competition on the number of cameras installed on a smartphone, and as Figure 7.2 illustrates, the penetration rate for multi-camera smartphones, particularly dual-camera ones, is rapidly increasing. This creates an excellent opportunity to explore computational imaging on mobile cameras, and develop algorithms for light field capture, compression, and rendering that are computationally efficient for the low-powered processors of the smartphones.

² <https://www.statista.com>

Bibliography

- [1] S. Hajisharif, J. Kronander, and J. Unger, “HDR Reconstruction for Alternating Gain (ISO) Sensor Readout,” in *Eurographics 2014 - Short Papers*, E. Galin and M. Wand, Eds. The Eurographics Association, 2014.
- [2] S. Hajisharif, J. Kronander, and J. Unger, “Adaptive DualISO HDR Reconstruction,” *EURASIP Journal on Image and Video Processing*, vol. 2015, no. 41, dec 2015.
- [3] E. Miandji, S. Hajisharif, and J. Unger, “A Unified Framework for Compression and Compressed Sensing of Light Fields and Light Field Videos,” *ACM Trans. Graph.*, vol. 38, no. 3, pp. 23:1–23:18, may 2019.
- [4] S. Hajisharif, E. Miandji, P. Larsson, K. Tran, and J. Unger, “Light field video compression and real time rendering,” *Computer Graphics Forum*, vol. 38, no. 7, pp. 265–276, 2019.
- [5] S. Hajisharif, E. Miandji, G. Baravadish, and J. Unger, “Compression and Real-Time Rendering of Inward Looking Spherical Light Fields,” in *Eurographics 2020 - Short Papers*, U. Assarsson and D. Panozzo, Eds. The Eurographics Association, (Submitted).
- [6] S. Hajisharif, E. Miandji, C. Guillemot, and J. Unger, “Single Sensor Compressive Light Field Video Camera,” in *Eurographics 2020*, U. Assarsson and D. Panozzo, Eds. The Eurographics Association, (Conditionally Accepted).
- [7] S. Hajisharif, J. Kronander, E. Miandji, and J. Unger, “Real-time Image Based Lighting with Streaming HDR Light Probe Sequences,” in *SIGRAD 2012*. Sweden: Linköping University Electronic Press, 2012. [pages 4 and 9]
- [8] J. Unger, J. Kronander, and S. Hajisharif, “Unified Reconstruction of Raw HDR Video Data,” in *High Dynamic Range Video: From Acquisition, to Display and Applications*, 1st ed., F. Dufaux, P. L. Callet, R. K. Mantiuk, and M. Mrak, Eds. Elsevier, 2016, ch. 2, pp. 62–83. [page xv]
- [9] J. Kronander, S. Gustavson, G. Bonnet, A. Ynnerman, and J. Unger, “A Unified Framework for Multi-Sensor HDR Video Reconstruction,” *Signal*

- Processing : Image Communications*, vol. 29, no. 2, pp. 203–215, 2014. [pages [xv](#), [9](#), [24](#), [26](#), [27](#), and [32](#)]
- [10] M. Levoy, R. Ng, A. Adams, M. Footer, and M. Horowitz, “Light field microscopy,” in *ACM SIGGRAPH 2006 Papers*, ser. SIGGRAPH ’06. New York, NY, USA: ACM, 2006, pp. 924–934. [pages [1](#) and [80](#)]
 - [11] N. Li, J. Ye, Y. Ji, H. Ling, and J. Yu, “Saliency detection on light field,” in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, June 2014, pp. 2806–2813. [page [1](#)]
 - [12] P. P. Srinivasan, M. W. Tao, R. Ng, and R. Ramamoorthi, “Oriented light-field windows for scene flow,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec 2015, pp. 3496–3504. [page [1](#)]
 - [13] S. W. Hasinoff, D. Sharlet, R. Geiss, A. Adams, J. T. Barron, F. Kainz, J. Chen, and M. Levoy, “Burst photography for high dynamic range and low-light imaging on mobile cameras,” *ACM Trans. Graph.*, vol. 35, no. 6, pp. 192:1–192:12, Nov. 2016. [page [2](#)]
 - [14] M. Elad and M. Aharon, “Image Denoising Via Sparse and Redundant Representations Over Learned Dictionaries,” *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, Dec 2006. [pages [2](#), [46](#), [48](#), and [56](#)]
 - [15] B. Wronski, I. Garcia-Dorado, M. Ernst, D. Kelly, M. Krainin, C.-K. Liang, M. Levoy, and P. Milanfar, “Handheld Multi-frame Super-resolution,” *ACM Trans. Graph.*, vol. 38, no. 4, pp. 28:1–28:18, jul 2019. [page [2](#)]
 - [16] N. Wadhwa, R. Garg, D. E. Jacobs, B. E. Feldman, N. Kanazawa, R. Carroll, Y. Movshovitz-Attias, J. T. Barron, Y. Pritch, and M. Levoy, “Synthetic depth-of-field with a single-camera mobile phone,” *ACM Trans. Graph.*, vol. 37, no. 4, pp. 64:1–64:13, Jul. 2018. [page [2](#)]
 - [17] X. Li, B. Gunturk, and L. , Zhang, “Image demosaicing: a systematic survey,” in *Visual Communications and Image Processing 2008*, vol. 6822, 2008. [pages [3](#) and [14](#)]
 - [18] S. Nayar and T. Mitsunaga, “High dynamic range imaging: Spatially varying pixel exposures,” in *CVPR*, 2000. [pages [3](#) and [27](#)]
 - [19] M. D. Tocci, C. Kiser, N. Tocci, and P. Sen, “A Versatile HDR Video Production System,” *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2011)*, vol. 30, no. 4, pp. 41:1–41:10, July 2011. [pages [3](#), [5](#), [26](#), [27](#), and [30](#)]
 - [20] J. Kronander, S. Gustavson, G. Bonnet, and J. Unger, “Unified hdr reconstruction from raw cfa data,” in *IEEE International Conference on Computational Photography (ICCP)*, April 2013, pp. 1–9. [pages [3](#), [30](#), [31](#), and [32](#)]

- [21] R. Ng, “Digital light field photography,” Ph.D. dissertation, Stanford, CA, USA, 2006, aAI3219345. [pages 3, 8, and 78]
- [22] K. Marwah, G. Wetzstein, Y. Bando, and R. Raskar, “Compressive Light Field Photography using Overcomplete Dictionaries and Optimized Projections,” *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 32, no. 4, pp. 1–11, 2013. [pages 3, 52, 81, 83, 86, 89, and 90]
- [23] E. Miandji, J. Unger, and C. Guillemot, “Multi-shot single sensor light field camera using a color coded mask,” in *26th European Signal Processing Conference (EUSIPCO) 2018*. IEEE, September 2018. [pages 3, 52, 81, 83, 84, 86, 87, 89, and 90]
- [24] H.-Y. Shum, S.-C. Chan, and S. B. Kang, *Image-based rendering*. Springer Science & Business Media, 2008. [pages 4 and 55]
- [25] P. Debevec, “Virtual cinematography: Relighting through computation,” *Computer*, vol. 39, no. 8, pp. 57–65, 2006. [page 4]
- [26] P. Debevec, “Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography,” in *ACM SIGGRAPH 2008 classes*. ACM, 2008, p. 32. [page 4]
- [27] S. Mangiat and J. Gibson, “Automatic scene relighting for video conferencing,” in *2009 16th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2009, pp. 2781–2784. [page 4]
- [28] E. Reinhard, G. Ward, S. Pattanaik, P. Debevec, and K. Myszkowski, *High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2010. [pages 4, 8, 18, 24, and 26]
- [29] J. Kronander, F. Banterle, A. Gardner, E. Miandji, and J. Unger, “Photorealistic rendering of mixed reality scenes,” *Computer Graphics Forum*, vol. 34, no. 2, pp. 643–665, 2015. [page 4]
- [30] T. Olson, “High dynamic range astronomical imaging,” in *Color and Imaging Conference*, vol. 2007, no. 1. Society for Imaging Science and Technology, 2007, pp. 304–312. [page 4]
- [31] B. Anconelli, M. Bertero, P. Boccacci, M. Carillet, and H. Lanteri, “Iterative methods for the reconstruction of astronomical images with high dynamic range,” *Journal of Computational and Applied Mathematics*, vol. 198, no. 2, pp. 321–331, 2007. [page 4]

- [32] A. Ramalli, M. Scaringella, G. Matrone, A. Dallai, E. Boni, A. S. Savoia, L. Bassi, G. E. Hine, and P. Tortoli, “High dynamic range ultrasound imaging with real-time filtered-delay multiply and sum beamforming,” in *2017 IEEE International Ultrasonics Symposium (IUS)*, Sep. 2017, pp. 1–4. [page 4]
- [33] E. Prado, I. Pavlov, N. Navab, and G. Zahnd, “Deep-learning high-dynamic-range ultrasound,” in *2019 IEEE International Ultrasonics Symposium (IUS)*. IEEE, 2019, pp. 479–482. [page 4]
- [34] I. M. Peters, C. Smit, J. J. Miller, and A. Lomako, “High dynamic range cmos-based mammography detector for ffdm and dbt,” in *Medical Imaging 2016: Physics of Medical Imaging*, vol. 9783. International Society for Optics and Photonics, 2016, p. 978316. [page 4]
- [35] M. A. Haidekker, L. D.-k. Morrison, A. Sharma, and E. Burke, “Enhanced dynamic range x-ray imaging,” *Computers in biology and medicine*, vol. 82, pp. 40–48, 2017. [page 4]
- [36] H. Seetzen, W. Heidrich, W. Stuerzlinger, G. Ward, L. Whitehead, M. Trentacoste, A. Ghosh, and A. Vorozcovs, “High dynamic range display systems,” *ACM transactions on graphics (TOG)*, vol. 23, no. 3, pp. 760–768, 2004. [page 4]
- [37] P. Lincoln, A. Blate, M. Singh, A. State, M. C. Whitton, T. Whitted, and H. Fuchs, “Scene-adaptive high dynamic range display for low latency augmented reality,” in *Proceedings of the 21st ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, 2017, pp. 1–7. [page 4]
- [38] C. W. Wyckoff, “An Experimental Extended Response Film,” Edgerton Germeshausen Grier, Inc, Boston, Massachusetts, Tech. Rep. Technical Report B-321, March 1961. [page 5]
- [39] A. Tomaszewska and R. Mantiuk, “Image registration for multi-exposure high dynamic range image acquisition,” 01 2007. [page 5]
- [40] S. Mann and R. Picard, “On being ‘undigital’ with digital cameras: extending dynamic range by combining differently exposed pictures,” *Proc. IS and T Annual Meeting*, vol. 48, 03 1996. [pages 5 and 25]
- [41] P. E. Debevec and J. Malik, “Recovering high dynamic range radiance maps from photographs,” in *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH ’97. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 1997, pp. 369–378. [pages 5, 25, 28, and 29]

- [42] F. E. Ives, “Parallax stereogram and process of making same,” USA, 1903. [page 6]
- [43] M. LIPPMANN, “Epreuves reversibles donnant la sensation du relief,” *J.de Phys*, vol. 7, pp. 821–825, 1908. [pages 6 and 80]
- [44] A. Gershun, “The light field,” *Journal of Mathematics and Physics*, vol. 18, no. 1–4, pp. 51–151, 1939. [page 6]
- [45] E. H. Adelson and J. R. Bergen, “The plenoptic function and the elements of early vision,” in *Computational Models of Visual Processing*. MIT Press, 1991, pp. 3–20. [pages 6 and 55]
- [46] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, “Light field photography with a hand-held plenoptic camera,” *Technical Report CTSR 2005-02*, vol. CTSR, 01 2005. [pages 6 and 80]
- [47] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy, “High performance imaging using large camera arrays,” in *ACM SIGGRAPH 2005 Papers*, ser. SIGGRAPH ’05. New York, NY, USA: ACM, 2005, pp. 765–776. [pages 6, 73, and 79]
- [48] J. Unger, S. Gustavson, M. Ollila, and M. Johannesson, “A real time light probe,” in *Eurographics Annual Conference 2004*, 2004. [pages 6 and 28]
- [49] P. Debevec, “Image-based lighting,” *IEEE Computer Graphics and Applications*, vol. 22, no. 2, pp. 26–34, March 2002. [pages 8 and 24]
- [50] E. A. Khan, “Image-based Material Editing,” Ph.D. dissertation, Orlando, FL, USA, 2006, aAI3242444. [page 8]
- [51] H.-Y. Shum, K.-T. Ng, and S.-C. Chan, “Virtual reality using the concentric mosaic: Construction, rendering and compression.” Institute of Electrical and Electronics Engineers, Inc., September 2000. [pages 8 and 79]
- [52] C. Buehler, M. Bosse, L. McMillan, S. Gortler, and M. Cohen, “Unstructured Lumigraph Rendering,” in *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH ’01. New York, NY, USA: ACM, 2001, pp. 425–432. [pages 8 and 82]
- [53] G. Wu, M. Zhao, L. Wang, Q. Dai, T. Chai, and Y. Liu, “Light field reconstruction using deep convolutional network on epi,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 1638–1646. [pages 8 and 71]

- [54] T.-C. Wang, J.-Y. Zhu, E. Hiroaki, M. Chandraker, A. Efros, and R. Ramamoorthi, “A 4d light-field dataset and cnn architectures for material recognition,” in *Proceedings of European Conference on Computer Vision (ECCV)*, 2016. [page 9]
- [55] T.-C. Wang, M. Chandraker, A. A. Efros, and R. Ramamoorthi, “Svbrdf-invariant shape and reflectance estimation from light-field cameras,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5451–5459. [pages 9 and 95]
- [56] H. Li, C. Guo, and S. Jia, “High-resolution light-field microscopy,” in *Frontiers in Optics 2017*. Optical Society of America, 2017. [page 9]
- [57] D. Tsai, D. G. Dansereau, T. Peynot, and P. Corke, “Image-based visual servoing with light field cameras,” *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 912–919, April 2017. [page 9]
- [58] N. Bedard, T. Shope, A. Hoberman, M. A. Haralam, N. Shaikh, J. Kovačević, N. Balram, and I. Tošić, “Light field otoscope design for 3d in vivo imaging of the middle ear,” *Biomed. Opt. Express*, vol. 8, no. 1, pp. 260–272, Jan 2017. [page 9]
- [59] B. E. Bayer, “Color imaging array,” Granted Patent US3 971 065A, 1976. [page 14]
- [60] E. B. Gindele and A. C. Gallagher, “Sparsely sampled image sensing device with color and luminance photosites,” nov 2002, uS Patent 6,476,865. [page 14]
- [61] M. Parmar and S. J. Reeves, “A perceptually based design methodology for color filter arrays [image reconstruction],” *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, pp. iii–473, 2004. [pages 14 and 94]
- [62] L. Condat, “A New Color Filter Array With Optimal Properties for Noiseless and Noisy Color Image Acquisition,” *IEEE Transactions on Image Processing*, vol. 20, no. 8, pp. 2200–*2210, Aug 2011. [page 14]
- [63] J. E. Adams Jr, “Interactions between color plane interpolation and other image processing functions in electronic photography,” in *Cameras and Systems for Electronic Photography and Scientific Imaging*, vol. 2416. International Society for Optics and Photonics, 1995, pp. 144–151. [page 14]
- [64] D. Kiku, Y. Monno, M. Tanaka, and M. Okutomi, “Residual interpolation for color image demosaicking,” in *2013 IEEE International Conference on Image Processing*. IEEE, 2013, pp. 2304–2308. [page 14]

- [65] Y. Monno, D. Kiku, M. Tanaka, and M. Okutomi, “Adaptive residual interpolation for color image demosaicking,” in *2015 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2015, pp. 3861–3865. [page 14]
- [66] A. Buades, B. Coll, J.-M. Morel, and C. Sbert, “Self-similarity driven color demosaicking,” *IEEE Transactions on Image Processing*, vol. 18, no. 6, pp. 1192–1202, 2009. [page 14]
- [67] J. Mairal, M. Elad, and G. Sapiro, “Sparse representation for color image restoration,” *IEEE Transactions on image processing*, vol. 17, no. 1, pp. 53–69, 2007. [page 14]
- [68] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, “Non-local sparse models for image restoration,” in *2009 IEEE 12th International Conference on Computer Vision*, Sept 2009, pp. 2272–2279. [page 14]
- [69] N.-S. Syu, Y.-S. Chen, and Y.-Y. Chuang, “Learning deep convolutional networks for demosaicing,” *arXiv preprint arXiv:1802.03769*, 2018. [page 14]
- [70] D. Menon and G. Calvagno, “Color image demosaicking: An overview,” *Signal Processing: Image Communication*, vol. 26, no. 8-9, pp. 518–533, 2011. [page 15]
- [71] S. W. Hasinoff, F. Durand, and W. T. Freeman, “Noise-optimal capture for high dynamic range photography,” in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2010, pp. 553–560. [pages 16, 29, and 32]
- [72] E. Reinhard, E. A. Khan, A. O. Akyz, and G. M. Johnson, *Color Imaging: Fundamentals and Applications*. Natick, MA, USA: A. K. Peters, Ltd., 2008. [page 17]
- [73] A. J. Theuwissen, *Solid-state imaging with charge-coupled devices*. Springer Science & Business Media, 2006, vol. 1. [page 17]
- [74] J. Hynecek, “Cds noise reduction of partially reset charge-detection nodes,” *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 49, no. 3, March 2002. [page 17]
- [75] R. Mancini, “Op amps for everyone,” 2002. [page 17]
- [76] C. Aguerrebere, J. Delon, Y. Gousseau, and P. Musé, “Study of the digital camera acquisition process and statistical modeling of the sensor raw data,” 2013. [pages 17 and 30]

- [77] I. Brouk, A. Nemirovsky, and Y. Nemirovsky, “Analysis of noise in cmos image sensor,” in *2008 IEEE International Conference on Microwaves, Communications, Antennas and Electronic Systems*, May 2008, pp. 1–8. [page 17]
- [78] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli *et al.*, “Image quality assessment: from error visibility to structural similarity,” *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004. [page 20]
- [79] L. Zhang, L. Zhang, X. Mou, and D. Zhang, “A comprehensive evaluation of full reference image quality assessment algorithms,” in *2012 19th IEEE International Conference on Image Processing*, Sep. 2012, pp. 1477–1480. [page 20]
- [80] T. O. Aydın, R. Mantiuk, and H.-P. Seidel, “Extending quality metrics to full luminance range images,” in *Human Vision and Electronic Imaging XIII*, vol. 6806. International Society for Optics and Photonics, 2008, p. 68060B. [pages 20, 21, and 42]
- [81] R. Mantiuk, K. J. Kim, A. G. Rempel, and W. Heidrich, “Hdr-vdp-2: A calibrated visual metric for visibility and quality predictions in all luminance conditions,” *ACM Transactions on graphics (TOG)*, vol. 30, no. 4, p. 40, 2011. [pages 20 and 21]
- [82] R. K. Mantiuk, K. Myszkowski, and H.-P. Seidel, *High Dynamic Range Imaging*. American Cancer Society, 2015, pp. 1–42. [page 21]
- [83] G. Eilertsen, R. K. Mantiuk, and J. Unger, “A high dynamic range video codec optimized by large-scale testing,” in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2016, pp. 1379–1383. [page 21]
- [84] V. Katkovnik, K. Egiazarian, and J. Astola, *Local approximation techniques in signal and image processing*. Signal Processing Institute, Tampere University of Technology: SPIE, 2006. [pages 24, 34, 37, and 41]
- [85] H. Seetzen, G. Ward, L. Whitehead, and W. Heidrich, “High Dynamic Range Display System,” in *ACM SIGGRAPH 2004 Emerging Technologies*, ser. SIGGRAPH ’04. New York, NY, USA: ACM, 2004. [page 24]
- [86] B. C. Madden, “Extended intensity range imaging,” *Technical Reports (CIS)*, p. 248, 1993. [page 24]
- [87] T. Mitsunaga and S. K. Nayar, “Radiometric self calibration,” in *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, vol. 1, June 1999, pp. 374–380 Vol. 1. [pages 25 and 28]

- [88] R. L. S. Mark A. Robertson, Sean Borman, “Estimation-theoretic approach to dynamic range enhancement using multiple exposures,” *Journal of Electronic Imaging*, vol. 12, no. 2, pp. 219 – 228 – 10, 2003. [pages 25 and 28]
- [89] T. Mertens, J. Kautz, and F. Van Reeth, “Exposure fusion: A simple and practical alternative to high dynamic range photography,” in *Computer graphics forum*, vol. 28, no. 1. Wiley Online Library, 2009, pp. 161–171. [page 25]
- [90] A. Tomaszewska and R. Mantiuk, “Image registration for multi-exposure high dynamic range image acquisition,” 2007. [page 25]
- [91] L. Bogoni, “Extending dynamic range of monochrome and color images through fusion,” in *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, vol. 3. IEEE, 2000, pp. 7–12. [page 25]
- [92] M. Gupta, D. Iso, and S. K. Nayar, “Fibonacci exposure bracketing for high dynamic range imaging,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1473–1480. [page 25]
- [93] O. Gallo, N. Gelfandz, W.-C. Chen, M. Tico, and K. Pulli, “Artifact-free high dynamic range imaging,” in *2009 IEEE International Conference on Computational Photography (ICCP)*. IEEE, 2009, pp. 1–7. [page 25]
- [94] H. Zimmer, A. Bruhn, and J. Weickert, “Freehand hdr imaging of moving scenes with simultaneous resolution enhancement,” in *Computer Graphics Forum*, vol. 30, no. 2. Wiley Online Library, 2011, pp. 405–414. [page 25]
- [95] C. Lee, Y. Li, and V. Monga, “Ghost-free high dynamic range imaging via rank minimization,” *IEEE Signal Processing Letters*, vol. 21, no. 9, pp. 1045–1049, 2014. [page 26]
- [96] T.-H. Oh, J.-Y. Lee, Y.-W. Tai, and I. S. Kweon, “Robust high dynamic range imaging by rank minimization,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 6, pp. 1219–1232, 2014. [page 26]
- [97] O. T. Tursun, A. O. Akyüz, A. Erdem, and E. Erdem, “The State of the Art in HDR Deghosting: A Survey and Evaluation,” *Computer Graphics Forum*, vol. 34, no. 2, pp. 683–707, 2015. [page 26]
- [98] S. Wu, J. Xu, Y.-W. Tai, and C.-K. Tang, “Deep high dynamic range imaging with large foreground motions,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 117–132. [page 26]
- [99] M. Aggarwal and N. Ahuja, “Split aperture imaging for high dynamic range,” *International Journal of Computer Vision*, vol. 58, no. 1, pp. 7–17, 2004. [pages 26 and 27]

- [100] A. Manakov, J. F. Restrepo, O. Klehm, R. Hegedüs, E. Eisemann, H.-P. Seidel, and I. Ihrke, “A reconfigurable camera add-on for high dynamic range, multi-spectral, polarization, and light-field imaging,” *ACM Trans. Graph. (Proc. SIGGRAPH 2013)*, vol. 32, no. 4, pp. 47:1–47:14, jul 2013. [page 27]
- [101] S. G. Narasimhan and S. K. Nayar, “Enhancing Resolution Along Multiple Imaging Dimensions Using Assorted Pixels,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 4, pp. 518–530, 2005. [pages 27 and 30]
- [102] M. Schoberl, A. Belz, J. Seiler, S. Foessel, and A. Kaup, “High dynamic range video by spatially non-regular optical filtering,” in *2012 19th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2012, pp. 2757–2760. [page 27]
- [103] C. Aguerrebere, A. Almansa, Y. Gousseau, J. Delon, and P. Musé, “Single Shot High Dynamic Range Imaging Using Piecewise Linear Estimators,” in *ICCP*, 2014. [page 27]
- [104] J. Tumblin, A. Agrawal, and R. Raskar, “Why i want a gradient camera,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, vol. 1, June 2005, pp. 103–110 vol. 1. [page 27]
- [105] S. K. Nayar, V. Branzoi, and T. E. Boult, “Programmable imaging using a digital micromirror array,” in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, vol. 1, June 2004, pp. I–I. [page 27]
- [106] S. K. Nayar, V. Branzoi, and T. E. Boult, “Programmable imaging: Towards a flexible camera,” *International Journal of Computer Vision*, vol. 70, no. 1, pp. 7–22, Oct 2006. [page 27]
- [107] S. Ri, M. Fujigaki, T. Matui, and Y. Morimoto, “Accurate pixel-to-pixel correspondence adjustment in a digital micromirror device camera by using the phase-shifting moiré method,” *Appl. Opt.*, vol. 45, no. 27, pp. 6940–6946, Sep 2006. [page 27]
- [108] S. K. Nayar and V. Branzoi, “Adaptive dynamic range imaging: optical control of pixel exposures over space and time,” in *Proceedings Ninth IEEE International Conference on Computer Vision*, Oct 2003, pp. 1168–1175 vol.2. [page 27]
- [109] J. Unger and S. Gustavson, “High-dynamic-range video for photometric measurement of illumination,” in *Sensors, Cameras, and Systems for Scientific/Industrial Applications VIII*, vol. 6501. International Society for Optics and Photonics, 2007, p. 65010E. [page 28]

- [110] J. Gu, Y. Hitomi, T. Mitsunaga, and S. Nayar, “Coded Rolling Shutter Photography: Flexible Space-Time Sampling,” in *IEEE International Conference on Computational Photography (ICCP)*, Mar 2010. [page 28]
- [111] A. Serrano, F. Heide, D. Gutierrez, G. Wetzstein, and B. Masia, “Convolutional Sparse Coding for High Dynamic Range Imaging,” in *Proceedings of the 37th Annual Conference of the European Association for Computer Graphics*, ser. EG '16. Goslar Germany, Germany: Eurographics Association, 2016, pp. 153–163. [page 28]
- [112] G. Eilertsen, J. Kronander, G. Denes, R. K. Mantiuk, and J. Unger, “Hdr image reconstruction from a single exposure using deep cnns,” *ACM Trans. Graph.*, vol. 36, no. 6, pp. 178:1–178:15, 2017. [page 28]
- [113] J. Cai, S. Gu, and L. Zhang, “Learning a deep single image contrast enhancer from multi-exposure images,” *IEEE Transactions on Image Processing*, vol. 27, no. 4, pp. 2049–2062, April 2018. [page 28]
- [114] alex, “Dynamic range improvement for canon dslrs with 8-channel sensor readout by alternating iso during sensor readout,” July 2013. [pages 28 and 36]
- [115] K. Kirk and H. J. Andersen, “Noise characterization of weighting schemes for combination of multiple exposures,” in *BMVC*, vol. 3. Citeseer, 2006, pp. 1129–1138. [page 28]
- [116] M. Granados, B. Ajdin, M. Wand, C. Theobalt, H. Seidel, and H. P. A. Lensch, “Optimal hdr reconstruction with linear digital cameras,” in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2010, pp. 215–222. [pages 28 and 32]
- [117] O. Gallo, M. Tico, R. Manduchi, N. Gelfand, and K. Pulli, “Metering for exposure stacks,” in *Computer Graphics Forum*, vol. 31, no. 2pt2. Wiley Online Library, 2012, pp. 479–488. [page 29]
- [118] E. Martinec, “Noise, Dynamic Range and Bit Depth in Digital SLRs,” 2008. [page 29]
- [119] R. Clark, “Digital camera sensor performance summary,” 2019. [page 29]
- [120] A. O. Akyüz and E. Reinhard, “Noise reduction in high dynamic range imaging,” *Journal of Visual Communication and Image Representation*, vol. 18, no. 5, pp. 366–376, 2007. [page 29]
- [121] R. Ramanath, W. E. Snyder, Y. Yoo, and M. S. Drew, “Color image processing pipeline,” *IEEE Signal Processing Magazine*, vol. 22, no. 1, pp. 34–43, Jan 2005. [page 29]

- [122] B. Ajdin, M. B. Hullin, C. Fuchs, H.-P. Seidel, and H. P. Lensch, “Demosaicing by smoothing along 1d features,” in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8. [page 30]
- [123] P. Chatterjee, N. Joshi, S. B. Kang, and Y. Matsushita, “Noise suppression in low-light images through joint denoising and demosaicing,” in *CVPR 2011*, June 2011, pp. 321–328. [page 30]
- [124] A. Joulin, S. Kang, and S. B. Kang, “Recovering stereo pairs from anaglyphs,” in *IEEE Conference on Computer Vision and Pattern Recognition*, IEEE. IEEE, June 2013, pp. 289–296. [page 30]
- [125] C. J. Schuler, M. Hirsch, S. Harmeling, and B. Schölkopf, “Non-stationary correction of optical aberrations,” in *2011 International Conference on Computer Vision*, Nov 2011, pp. 659–666. [page 30]
- [126] C. J. Schuler, H. C. Burger, S. Harmeling, and B. Schölkopf, “A machine learning approach for non-blind image deconvolution,” in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, June 2013, pp. 1067–1074. [page 30]
- [127] F. Heide, M. Steinberger, Y.-T. Tsai, M. Rouf, D. Pająk, D. Reddy, O. Gallo, J. Liu, W. Heidrich, K. Egiazarian *et al.*, “Flexisp: A flexible camera image processing framework,” *ACM Transactions on Graphics (TOG)*, vol. 33, no. 6, p. 231, 2014. [page 30]
- [128] H. Takeda, S. Farsiu, and P. Milanfar, “Kernel Regression for Image Processing and Reconstruction,” *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 349–366, Feb 2007. [pages 31, 34, 35, 40, 42, and 44]
- [129] H. Knutsson and C. Westin, “Normalized and differential convolution,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, June 1993, pp. 515–523. [page 31]
- [130] C. Tomasi and R. Manduchi, “Bilateral filtering for gray and color images,” in *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, Jan 1998, pp. 839–846. [page 31]
- [131] C. Loader, *Local regression and likelihood*. Springer Science & Business Media, 2006. [page 31]
- [132] P. Milanfar, “A Tour of Modern Image Filtering: New Insights and Methods, Both Practical and Theoretical,” *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 106–128, Jan 2013. [page 31]

- [133] C. Aguerrebere, J. Delon, Y. Gousseau, and P. Musé, “Best algorithms for hdr image generation. a study of performance bounds,” *SIAM Journal on Imaging Sciences*, vol. 7, no. 1, pp. 1–34, 2014. [page 32]
- [134] A. Foi, “Clipped noisy images: Heteroskedastic modeling and practical denoising,” *Signal Processing*, vol. 89, no. 12, pp. 2609–2629, 2009. [page 32]
- [135] R. Tibshirani and T. Hastie, “Local Likelihood Estimation,” *Journal of the American Statistical Association*, vol. 82, no. 398, pp. 559–567, 1987. [page 35]
- [136] L. Stankovic, “Performance analysis of the adaptive algorithm for bias-to-variance tradeoff,” *Signal Processing, IEEE Transactions on*, vol. 52, no. 5, pp. 1228–1234, May 2004. [page 42]
- [137] M. Narwaria, R. Mantiuk, M. P. Da Silva, and P. Le Callet, “Hdr-vdp-2.2: a calibrated method for objective quality prediction of high-dynamic range and standard images,” *Journal of Electronic Imaging*, vol. 24, no. 1, p. 010501, 2015. [page 42]
- [138] H. Takeda, P. Milanfar, M. Protter, and M. Elad, “Super-Resolution Without Explicit Subpixel Motion Estimation,” *IEEE Transactions on Image Processing*, vol. 18, no. 9, pp. 1958–1975, Sep. 2009. [page 43]
- [139] S. Mallat, *A Wavelet Tour of Signal Processing: The Sparse Way*. Elsevier Science, 2008. [pages 46 and 57]
- [140] E. J. Candès and D. L. Donoho, “Ridgelets: A key to higher-dimensional intermittency?” *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 357, no. 1760, pp. 2495–2509, 1999. [page 46]
- [141] J.-L. Starck, E. J. Candes, and D. L. Donoho, “The curvelet transform for image denoising,” *IEEE Transactions on Image Processing*, vol. 11, no. 6, pp. 670–684, June 2002. [page 46]
- [142] J.-L. Starck, F. Murtagh, and J. Fadili, *Sparse image and signal processing: Wavelets and related geometric multiscale analysis*. Cambridge university press, 2015. [page 46]
- [143] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, “Online learning for matrix factorization and sparse coding,” *J. Mach. Learn. Res.*, vol. 11, pp. 19–60, March 2010. [pages 46, 48, 49, and 86]
- [144] G. Peyré, J. Fadili, and J.-L. Starck, “Learning the morphological diversity,” *SIAM Journal on Imaging Sciences*, vol. 3, no. 3, pp. 646–669, 2010. [pages 46 and 48]

- [145] S. S. Chen, D. L. Donoho, and M. A. Saunders, “Atomic decomposition by basis pursuit,” *SIAM Review*, vol. 43, no. 1, pp. 129–159, Jan 2001. [page 47]
- [146] E. Elhamifar and R. Vidal, “Robust classification using structured sparse representation,” in *2011 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2011, pp. 1873–1879. [page 47]
- [147] D. L. Donoho and J. Tanner, “Sparse nonnegative solution of underdetermined linear equations by linear programming,” *Proceedings of the National Academy of Sciences (PNAS)*, vol. 102, no. 27, pp. 9446–9451, 2005. [page 47]
- [148] C. You and R. Vidal, “Geometric conditions for subspace-sparse recovery,” in *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ser. ICML’15. JMLR, 2015, pp. 1585–1593. [page 47]
- [149] S. G. Mallat and Z. Zhang, “Matching pursuits with time-frequency dictionaries,” *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, Dec 1993. [page 47]
- [150] Y. C. Pati, R. Rezaifar, and P. S. Krishnaprasad, “Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition,” in *Proceedings of 27th Asilomar Conference on Signals, Systems and Computers*, Nov 1993, pp. 40–44. [pages 47 and 89]
- [151] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, “Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 586–597, Dec 2007. [page 47]
- [152] E. van den Berg and M. P. Friedlander, “Probing the pareto frontier for basis pursuit solutions,” *SIAM Journal on Scientific Computing*, vol. 31, no. 2, pp. 890–912, Nov 2009. [page 47]
- [153] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, “Sparse reconstruction by separable approximation,” *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 2479–2493, July 2009. [pages 47 and 89]
- [154] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, “Least angle regression,” *Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004. [page 47]
- [155] Z. Zhang, Y. Xu, J. Yang, X. Li, and D. Zhang, “A survey of sparse representation: Algorithms and applications,” *IEEE Access*, vol. 3, pp. 490–530, May 2015. [page 47]

- [156] E. Miandji, M. Emadi, J. Unger, and E. Afshari, “On Probability of Support Recovery for Orthogonal Matching Pursuit Using Mutual Coherence,” *IEEE Signal Processing Letters*, vol. 24, no. 11, pp. 1646–1650, Nov 2017. [pages 47, 67, and 75]
- [157] A. Mehranian, H. S. Rad, A. Rahmim, M. R. Ay, and H. Zaidi, “Smoothly clipped absolute deviation (scad) regularization for compressed sensing mri using an augmented lagrangian scheme,” *Magnetic Resonance Imaging*, vol. 31, no. 8, pp. 1399–1411, 2013. [page 47]
- [158] Y.-R. Fan, A. Buccini, M. Donatelli, and T.-Z. Huang, “A non-convex regularization approach for compressive sensing,” *Advances in Computational Mathematics*, Aug 2018. [page 47]
- [159] C.-H. Zhang, “Nearly unbiased variable selection under minimax concave penalty,” *The Annals of Statistics*, vol. 38, no. 2, pp. 894–942, April 2010. [page 47]
- [160] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, “A fast approach for overcomplete sparse decomposition based on smoothed ℓ^0 norm,” *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 289–301, Jan 2009. [pages 47 and 89]
- [161] E. J. Candès, X. Li, Y. Ma, and J. Wright, “Robust principal component analysis?” *Journal of the ACM*, vol. 58, no. 3, pp. 11:1–11:37, Jun 2011. [page 47]
- [162] V. de Silva and L. Lim, “Tensor rank and the ill-posedness of the best low-rank approximation problem,” *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 3, pp. 1084–1127, Sept 2008. [page 47]
- [163] T. Lin and H. Zha, “Riemannian manifold learning,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 5, pp. 796–809, May 2008. [page 47]
- [164] L. v. d. Maaten and G. Hinton, “Visualizing data using t-sne,” *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008. [page 47]
- [165] M. Aharon, M. Elad, and A. Bruckstein, “K-svd: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, Nov 2006. [pages 48, 49, 58, 59, and 65]
- [166] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, “A multilinear singular value decomposition,” *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, Mar 2000. [page 48]

- [167] R. Rubinstein, A. M. Bruckstein, and M. Elad, “Dictionaries for sparse representation modeling,” *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1045–1057, 2010. [page 49]
- [168] E. J. Candes and M. B. Wakin, “An introduction to compressive sampling,” *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, March 2008. [page 50]
- [169] G. K. Wallace, “The jpeg still picture compression standard,” *IEEE transactions on consumer electronics*, vol. 38, no. 1, 1992. [page 50]
- [170] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*. Springer New York, 2013. [page 50]
- [171] E. J. Candès, J. K. Romberg, and T. Tao, “Stable signal recovery from incomplete and inaccurate measurements,” *Communications on Pure and Applied Mathematics*, vol. 59, no. 8, pp. 1207–1223, 2006. [pages 51 and 57]
- [172] E. J. Candès, M. B. Wakin, and S. P. Boyd, “Enhancing sparsity by reweighted ℓ_1 minimization,” *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, Dec 2008. [page 51]
- [173] D. L. Donoho and M. Elad, “Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ_1 minimization,” *Proceedings of the National Academy of Sciences*, vol. 100, no. 5, pp. 2197–2202, 2003. [page 51]
- [174] X. Li, J. Ye, G. Li, X. He, and H. Bai, “Sensing matrix optimization for compressed sensing with consideration of sparse representation error and rank constraint,” in *2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, June 2015, pp. 93–98. [page 51]
- [175] L. Yu, H. Bai, and X. Wan, “Optimal sensing matrix for compressed sensing,” in *2011 International Conference on Electronics, Communications and Control (ICECC)*, Sep. 2011, pp. 360–363. [page 51]
- [176] Y. Inagaki, Y. Kobayashi, K. Takahashi, T. Fujii, and H. Nagahara, “Learning to capture light fields through a coded aperture camera,” in *Computer Vision – ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham: Springer International Publishing, 2018, pp. 431–448. [pages 51 and 90]
- [177] R. Baraniuk, “A lecture on compressive sensing,” *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118–121, 2007. [page 51]

- [178] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 83–91, March 2008. [page 52]
- [179] O. Nabati, D. Mendlovic, and R. Giryes, "Fast and accurate reconstruction of compressed color light field," in *2018 IEEE International Conference on Computational Photography (ICCP)*, May 2018, pp. 1–11. [pages 52, 81, and 85]
- [180] G. Huang, H. Jiang, K. Matthews, and P. Wilford, "Lensless imaging by compressive sensing," in *2013 IEEE International Conference on Image Processing*. IEEE, 2013, pp. 2101–2105. [pages 52 and 81]
- [181] R. G. Baraniuk, T. Goldstein, A. C. Sankaranarayanan, C. Studer, A. Veeraraghavan, and M. B. Wakin, "Compressive video sensing: Algorithms, architectures, and applications," *IEEE Signal Processing Magazine*, vol. 34, no. 1, pp. 52–66, Jan 2017. [page 52]
- [182] R. F. Marcia and R. M. Willett, "Compressive coded aperture video reconstruction," in *2008 16th European Signal Processing Conference*. IEEE, 2008, pp. 1–5. [page 52]
- [183] N. Radwell, K. J. Mitchell, G. M. Gibson, M. P. Edgar, R. Bowman, and M. J. Padgett, "Single-pixel infrared and visible microscope," *Optica*, vol. 1, no. 5, pp. 285–289, 2014. [page 52]
- [184] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse mri: The application of compressed sensing for rapid mr imaging," *Magnetic Resonance in Medicine*, vol. 58, no. 6, pp. 1182–1195, Oct 2007. [page 52]
- [185] J. Trzasko and A. Manduca, "Highly undersampled magnetic resonance image reconstruction via homotopic ℓ_0 -minimization," *IEEE Transactions on Medical imaging*, vol. 28, no. 1, pp. 106–121, 2008. [page 52]
- [186] H. Yu and G. Wang, "Compressed sensing based interior tomography," *Physics in medicine & biology*, vol. 54, no. 9, p. 2791, 2009. [page 52]
- [187] M. A. Sheikh, O. Milenkovic, and R. G. Baraniuk, "Designing compressive sensing dna microarrays," in *2007 2nd IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*. IEEE, 2007, pp. 141–144. [page 52]
- [188] K. Egiazarian, A. Foi, and V. Katkovnik, "Compressed sensing image reconstruction via recursive spatially adaptive filtering," in *2007 IEEE International Conference on Image Processing*, vol. 1. IEEE, 2007, pp. I–549. [page 52]

- [189] M. Levoy and P. Hanrahan, “Light field rendering,” in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH ’96. New York, NY, USA: ACM, 1996, pp. 31–42. [pages 55 and 82]
- [190] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, “The lumigraph,” in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH ’96. New York, NY, USA: ACM, 1996, pp. 43–54. [pages 55 and 73]
- [191] I. Ihm, S. Park, and R. K. Lee, “Rendering of spherical light fields,” in *Proceedings The Fifth Pacific Conference on Computer Graphics and Applications*. IEEE, 1997, pp. 59–68. [page 55]
- [192] E. Camahort, A. Leros, and D. Fussell, “Uniformly sampled light fields,” in *Rendering Techniques 98*. Springer, 1998, pp. 117–130. [page 55]
- [193] E. Camahort, *4D light-field modeling and rendering*. Computer Science Department, University of Texas at Austin, 2001. [page 55]
- [194] E. Miandji, J. Kronander, and J. Unger, “Compressive image reconstruction in reduced union of subspaces,” *Comput. Graph. Forum*, vol. 34, no. 2, pp. 33–44, May 2015. [pages 56, 57, and 58]
- [195] J. Zepeda, C. Guillemot, and E. Kijak, “Image compression using sparse representations and the iteration-tuned and aligned dictionary,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 5, pp. 1061–1073, Sep. 2011. [page 57]
- [196] E. Miandji, J. Kronander, and J. Unger, “Learning based compression of surface light fields for real-time rendering of global illumination scenes,” in *SIGGRAPH Asia 2013 Technical Briefs*, ser. SA ’13. New York, NY, USA: ACM, 2013, pp. 24:1–24:4. [pages 57 and 63]
- [197] E. J. Candes and T. Tao, “Near-optimal signal recovery from random projections: Universal encoding strategies?” *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406–5425, Dec 2006. [page 57]
- [198] S. Lloyd, “Least squares quantization in pcm,” *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982. [pages 57 and 62]
- [199] P.-P. Sloan, J. Hall, J. Hart, and J. Snyder, “Clustered principal components for precomputed radiance transfer,” *ACM Trans. Graph.*, vol. 22, pp. 382–391, 2003. [page 57]

- [200] M. Guthe, G. Müller, M. Schneider, and R. Klein, “Btf-cielab: A perceptual difference measure for quality assessment and compression of btfs,” *Computer Graphics Forum*, vol. 28, no. 1, pp. 101–113, 2009. [page 58]
- [201] Y.-T. Tsai, “Multiway k-clustered tensor approximation: Toward high-performance photorealistic data-driven rendering,” *ACM Transactions on Graphics*, vol. 34, no. 5, pp. 157:1–157:15, Nov 2015. [page 58]
- [202] E. Elhamifar and R. Vidal, “Sparse subspace clustering: Algorithm, theory, and applications,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2765–2781, Nov 2013. [page 58]
- [203] M. Soltanolkotabi, E. J. Candes *et al.*, “A geometric analysis of subspace clustering with outliers,” *The Annals of Statistics*, vol. 40, no. 4, pp. 2195–2238, 2012. [page 58]
- [204] Y. C. Eldar, P. Kuppinger, and H. Bolcskei, “Block-sparse signals: Uncertainty relations and efficient recovery,” *IEEE Transactions on Signal Processing*, vol. 58, no. 6, pp. 3042–3054, June 2010. [page 58]
- [205] O. Bryt and M. Elad, “Compression of facial images using the k-svd algorithm,” *J. Vis. Comun. Image Represent.*, vol. 19, pp. 270–282, 2008. [page 59]
- [206] M. Rahmani and G. K. Atia, “Coherence pursuit: Fast, simple, and robust principal component analysis,” *IEEE Transactions on Signal Processing*, vol. 65, no. 23, pp. 6260–6275, Dec 2017. [page 62]
- [207] N. Sabater, G. Boisson, B. Vandame, P. Kerbiriou, F. Babon, M. Hog, R. Gendrot, T. Langlois, O. Bureller, A. Schubert, and V. Allié, “Dataset and pipeline for multi-view light-field video,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, July 2017, pp. 1743–1753. [pages 64, 66, and 68]
- [208] A. Cohen, I. Daubechies, and J. Feauveau, “Biorthogonal bases of compactly supported wavelets,” *Communications on Pure and Applied Mathematics*, vol. 45, no. 5, pp. 485–560, June 1992. [page 65]
- [209] D. Taubman and M. Marcellin, *JPEG2000 Image Compression Fundamentals, Standards and Practice*. Springer Publishing Company, Incorporated, 2013. [page 65]
- [210] Z. Ben-Haim, Y. C. Eldar, and M. Elad, “Coherence-Based Performance Guarantees for Estimating a Sparse Vector Under Random Noise,” *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5030–5043, Oct 2010. [pages 66 and 67]

- [211] L. Guillo, X. jiang, G. Lafruit, and C. Guillemot, “Light field video dataset captured by a r8 raytrix camera,” April 2018. [pages 66, 67, 68, 87, 89, and 91]
- [212] V. Abolghasemi, S. Ferdowsi, B. Makkiabadi, and S. Sanei, “On optimization of the measurement matrix for compressive sensing,” in *2010 18th European Signal Processing Conference*, Aug 2010, pp. 427–431. [page 66]
- [213] D. Arthur and S. Vassilvitskii, “K-means++: The Advantages of Careful Seeding,” in *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA ’07. Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035. [page 67]
- [214] K. Zhang, W. Zuo, and L. Zhang, “Ffdnet: Toward a fast and flexible solution for cnn-based image denoising,” *IEEE Transactions on Image Processing*, vol. 27, no. 9, pp. 4608–4622, Sep. 2018. [pages 67 and 68]
- [215] N. K. Kalantari, T.-C. Wang, and R. Ramamoorthi, “Learning-based View Synthesis for Light Field Cameras,” *ACM Trans. Graph.*, vol. 35, no. 6, pp. 193:1–193:10, nov 2016. [page 71]
- [216] B. Mildenhall, P. P. Srinivasan, R. Ortiz-Cayon, N. K. Kalantari, R. Ramamoorthi, R. Ng, and A. Kar, “Local light field fusion: Practical view synthesis with prescriptive sampling guidelines,” *ACM Trans. Graph.*, vol. 38, no. 4, Jul. 2019. [pages 71 and 82]
- [217] M. DuVall, J. Flynn, M. Broxton, and P. Debevec, “Compositing Light Field Video Using Multiplane Images,” in *ACM SIGGRAPH 2019 Posters*, ser. SIGGRAPH ’19. New York, NY, USA: ACM, 2019, pp. 67:1–67:2. [pages 71 and 78]
- [218] M. Bemana, K. Myszkowski, H.-P. Seidel, and T. Ritschel, “Neural View-Interpolation for Sparse LightField Video,” *arXiv preprint arXiv:1910.13921*, 2019. [page 71]
- [219] W. D. Fisher, “On grouping for maximum homogeneity,” *Journal of the American Statistical Association*, vol. 53, no. 284, pp. 789–798, 1958. [page 73]
- [220] E. J. Candès, “The restricted isometry property and its implications for compressed sensing,” *Comptes Rendus Mathématique*, vol. 346, no. 9, pp. 589–592, May 2008. [page 75]
- [221] D. Donoho, M. Elad, and V. Temlyakov, “Stable recovery of sparse overcomplete representations in the presence of noise,” *Information Theory, IEEE Transactions on*, vol. 52, no. 1, pp. 6–18, Jan 2006. [page 75]

- [222] Y. C. Eldar and M. Mishali, “Robust recovery of signals from a structured union of subspaces,” *IEEE Transactions on Information Theory*, vol. 55, no. 11, pp. 5302–5316, Nov 2009. [page 75]
- [223] A. Smolic, K. Müller, P. Merkle, M. Kautzner, and T. Wiegand, “3d video objects for interactive applications,” in *2005 13th European Signal Processing Conference*, Sep. 2005, pp. 1–4. [page 78]
- [224] A. Kubota, A. Smolic, M. Magnor, M. Tanimoto, T. Chen, and C. Zhang, “Multi-view imaging and 3dtv.” *IEEE*, November 2007. [page 78]
- [225] G. Wetzstein, D. Lanman, M. Hirsch, and R. Raskar, “Tensor Displays: Compressive Light Field Synthesis using Multilayer Displays with Directional Backlighting,” *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 31, no. 4, pp. 1–11, 2012. [page 78]
- [226] A. Jones, J. Unger, K. Nagano, J. Busch, X. Yu, H.-Y. Peng, O. Alexander, M. Bolas, and P. Debevec, “An Automultiscopic Projector Array for Interactive Digital Humans,” in *SIGGRAPH 2015*. Los Angeles, CA: ACM Press, aug 2015, pp. 1–1. [page 78]
- [227] H.-G. Jeon, J. Park, G. Choe, J. Park, Y. Bok, Y.-W. Tai, and I. So Kweon, “Accurate Depth Map Estimation From a Lenslet Light Field Camera,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. [page 78]
- [228] O. Johannsen, K. Honauer, B. Goldluecke, A. Alperovich, F. Battisti, Y. Bok, M. Brizzi, M. Carli, G. Choe, M. Diebold, M. Gutsche, H. Jeon, I. S. Kweon, J. Park, J. Park, H. Schilling, H. Sheng, L. Si, M. Strecke, A. Sulc, Y. Tai, Q. Wang, T. Wang, S. Wanner, Z. Xiong, J. Yu, S. Zhang, and H. Zhu, “A Taxonomy and Evaluation of Dense Light Field Depth Estimation Algorithms,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, July 2017, pp. 1795–1812. [page 78]
- [229] S. Heber and T. Pock, “Convolutional networks for shape from light field,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3746–3754. [page 78]
- [230] A. Mustafa, M. Volino, J. Guillemaut, and A. Hilton, “4d temporally coherent light-field video,” in *2017 International Conference on 3D Vision (3DV)*, Oct 2017, pp. 29–37. [page 78]
- [231] H. Zhu, Q. Zhang, and Q. Wang, “4d light field superpixel and segmentation,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. [page 78]

- [232] D. G. Dansereau, G. Schuster, J. Ford, and G. Wetzstein, “A Wide-Field-Of-View Monocentric Light Field Camera,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. [page 78]
- [233] D. Antensteiner and S. Stolc, “Full brdf reconstruction using cnns from partial photometric stereo-light field data,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 1726–1734. [page 78]
- [234] P. Debevec, “The light stages and their applications to photoreal digital actors,” *SIGGRAPH Asia*, vol. 2, no. 4, 2012. [page 78]
- [235] T. Fujii, K. Mori, K. Takeda, K. Mase, M. Tanimoto, and Y. Suenaga, “Multipoint Measuring System for Video and Sound - 100-camera and microphone system,” in *2006 IEEE International Conference on Multimedia and Expo*, July 2006, pp. 437–440. [pages 79 and 80]
- [236] C. Zhang and T. Chen, “A Self-Reconfigurable Camera Array,” in *Eurographics Workshop on Rendering*, A. Keller and H. W. Jensen, Eds. The Eurographics Association, 2004. [page 79]
- [237] F. Marton, E. Gobbetti, F. Bettio, J. A. Iglesias Guitián, and R. Pintus, “A real-time coarse-to-fine multiview capture system for all-in-focus rendering on a light-field display,” in *2011 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON)*, May 2011, pp. 1–4. [page 79]
- [238] K. Venkataraman, D. Lelescu, J. Duparré, A. McMahon, G. Molina, P. Chatterjee, R. Mullis, and S. Nayar, “Picam: An ultra-thin high performance monolithic camera array,” *ACM Trans. Graph.*, vol. 32, no. 6, pp. 166:1–166:13, nov 2013. [page 79]
- [239] A. Jones, J. Unger, K. Nagano, J. Busch, X. Yu, H.-Y. Peng, J. Barreto, O. Alexander, M. Bolas, and P. Debevec, “Time-offset conversations on a life-sized automultiscopic projector array,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2016. [page 79]
- [240] R. S. Overbeck, D. Erickson, D. Evangelakos, M. Pharr, and P. Debevec, “A system for acquiring, processing, and rendering panoramic light field stills for virtual reality,” in *SIGGRAPH Asia 2018 Technical Papers*. ACM, 2018, p. 197. [page 80]
- [241] R. Yang, G. Welch, and G. Bishop, “Real-time consensus-based scene reconstruction using commodity graphics hardware,” in *10th Pacific Conference on Computer Graphics and Applications, 2002. Proceedings.*, Oct 2002, pp. 225–234. [page 80]

- [242] T. Basha, S. Avidan, A. Hornung, and W. Matusik, “Structure and motion from scene registration,” in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 1426–1433. [page 80]
- [243] E. H. Adelson and J. Y. A. Wang, “Single lens stereo with a plenoptic camera,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 99–106, Feb 1992. [page 80]
- [244] T. Georgeiv, K. C. Zheng, B. Curless, D. Salesin, S. Nayar, and C. Intwala, “Spatio-Angular Resolution Tradeoffs in Integral Photography,” in *Symposium on Rendering*, T. Akenine-Moeller and W. Heidrich, Eds. The Eurographics Association, 2006. [page 80]
- [245] R. Raskar, A. Agrawal, and J. Tumblin, “Coded Exposure Photography: Motion Deblurring Using Fluttered Shutter,” *ACM Trans. Graph.*, vol. 25, no. 3, pp. 795–804, Jul. 2006. [page 81]
- [246] E. E. Fenimore and T. M. Cannon, “Coded aperture imaging with uniformly redundant arrays,” *Applied Optics*, vol. 17, no. 3, pp. 337–347, Feb 1978. [page 81]
- [247] G. K. Skinner, “X-ray imaging with coded masks,” *Scientific American*, vol. 259, no. 2, pp. 84–89, 1988. [page 81]
- [248] S. R. Gottesman and E. Fenimore, “New family of binary arrays for coded aperture imaging,” *Applied optics*, vol. 28, no. 20, pp. 4344–4352, 1989. [page 81]
- [249] A. Veeraraghavan, R. Raskar, A. Agrawal, A. Mohan, and J. Tumblin, “Dappled photography: Mask enhanced cameras for heterodyned light fields and coded aperture refocusing,” *ACM Transactions on Graphics*, vol. 26, no. 3, Jul 2007. [page 81]
- [250] C.-K. Liang, T.-H. Lin, B.-Y. Wong, C. Liu, and H. H. Chen, “Programmable Aperture Photography: Multiplexed Light Field Acquisition,” in *ACM SIGGRAPH 2008 Papers*, ser. SIGGRAPH ’08. New York, NY, USA: ACM, 2008, pp. 55:1–55:10. [page 81]
- [251] Z. Xu and E. Y. Lam, “A high-resolution lightfield camera with dual-mask design,” pp. 1–11, 2012. [page 81]
- [252] S. D. Babacan, R. Ansorge, M. Luessi, P. R. Mataran, R. Molina, and A. K. Katsaggelos, “Compressive Light Field Sensing,” *IEEE Transactions on Image Processing*, vol. 21, no. 12, pp. 4746–4757, Dec 2012. [page 81]

- [253] J. Unger, A. Wenger, T. Hawkins, A. Gardner, and P. Debevec, “Capturing and rendering with incident light fields,” in *Proceedings of the 14th Eurographics Workshop on Rendering*. Eurographics Association, 2003, pp. 141–149. [page 82]
- [254] S. Ray, *Applied Photographic Optics: Lenses and optical systems for photography, film, video, electronic and digital imaging*, 02 2002. [page 83]
- [255] E. Miandji, *Sparse Representation of Visual Data for Compression and Compressed Sensing*, ser. Linköping Studies in Science and Technology. Dissertations. Linköping University Electronic Press, 2018. [page 85]
- [256] S. Niklaus and F. Liu, “Context-aware synthesis for video frame interpolation,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2018, pp. 1701–1710. [page 89]
- [257] D. Needell and R. Vershynin, “Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit,” *Selected Topics in Signal Processing, IEEE Journal of*, vol. 4, no. 2, pp. 310–316, April 2010. [page 89]
- [258] D. Needell and J. Tropp, “Cosamp: Iterative signal recovery from incomplete and inaccurate samples,” *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009. [page 89]
- [259] M. A. Khajehnejad, W. Xu, A. S. Avestimehr, and B. Hassibi, “Improving the thresholds of sparse recovery: An analysis of a two-step reweighted basis pursuit algorithm,” *IEEE Transactions on Information Theory*, vol. 61, no. 9, pp. 5116–5128, Sept 2015. [page 89]
- [260] P. Sen, N. K. Kalantari, M. Yaesoubi, S. Darabi, D. B. Goldman, and E. Shechtman, “Robust patch-based hdr reconstruction of dynamic scenes,” *ACM Trans. Graph.*, vol. 31, no. 6, pp. 203:1–203:11, nov 2012. [page 94]
- [261] N. K. Kalantari and R. Ramamoorthi, “Deep high dynamic range imaging of dynamic scenes,” *ACM Trans. Graph.*, vol. 36, no. 4, pp. 144–1, 2017. [page 94]
- [262] I. Choi, S. Baek, and M. H. Kim, “Reconstructing interlaced high-dynamic-range video using joint learning,” *IEEE Transactions on Image Processing*, vol. 26, no. 11, pp. 5353–5366, Nov 2017. [page 94]
- [263] R. Mantiuk, A. Efremov, K. Myszkowski, and H.-P. Seidel, “Backward compatible high dynamic range mpeg video compression,” *ACM Transactions on Graphics (TOG)*, vol. 25, no. 3, pp. 713–723, 2006. [page 95]

Publications

Publications

The publications associated with this thesis have been removed for copyright reasons. For more details about these see:

<http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-163693>

FACULTY OF SCIENCE AND ENGINEERING

Linköping Studies in Science and Technology, Dissertation No. 2046, 2020
Department of Science and Technology

Linköping University
SE-581 83 Linköping, Sweden

www.liu.se