Linköping studies in science and technology.
Dissertation, No. 1963

SPARSE REPRESENTATION OF VISUAL DATA FOR
COMPRESSION AND COMPRESSED SENSING

Ehsan Miandji

Division of Media and Information Technology
Department of Science and Technology
Linköping University, SE-601 74 Norrköping, Sweden
Norrköping, December 2018

Description of the cover image

The cover of this thesis depicts the sparse representation of a light field. From the back side to front, a light field is divided into small elements, then these elements (signals) are projected onto an ensemble of two 5D dictionaries to produce a sparse coefficient vector. I hope that the reader can deduce where the dimensionality of those Rubik's Cube looking matrices come from after reading Chapter 3 of this thesis. The light field used on the cover is a part of the Stanford Lego Gantry database (<http://lightfield.stanford.edu/>)

Sparse Representation of Visual Data for Compression and Compressed Sensing

Copyright © 2018 Ehsan Miandji (unless otherwise noted)

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

ISBN: 978-91-7685-186-9

ISSN: 0345-7524

Printed in Sweden by LiU-Tryck, Linköping, 2018

Abstract

The ongoing advances in computational photography have introduced a range of new imaging techniques for capturing multidimensional visual data such as light fields, BRDFs, BTFs, and more. A key challenge inherent to such imaging techniques is the large amount of high dimensional visual data that is produced, often requiring GBs, or even TBs, of storage. Moreover, the utilization of these datasets in real time applications poses many difficulties due to the large memory footprint. Furthermore, the acquisition of large-scale visual data is very challenging and expensive in most cases. This thesis makes several contributions with regards to acquisition, compression, and real time rendering of high dimensional visual data in computer graphics and imaging applications.

Contributions of this thesis reside on the strong foundation of sparse representations. Numerous applications are presented that utilize sparse representations for compression and compressed sensing of visual data. Specifically, we present a single sensor light field camera design, a compressive rendering method, a real time precomputed photorealistic rendering technique, light field (video) compression and real time rendering, compressive BRDF capture, and more. Another key contribution of this thesis is a general framework for compression and compressed sensing of visual data, regardless of the dimensionality. As a result, any type of discrete visual data with arbitrary dimensionality can be captured, compressed, and rendered in real time.

This thesis makes two theoretical contributions. In particular, uniqueness conditions for recovering a sparse signal under an ensemble of multidimensional dictionaries is presented. The theoretical results discussed here are useful for designing efficient capturing devices for multidimensional visual data. Moreover, we derive the probability of successful recovery of a noisy sparse signal using OMP, one of the most widely used algorithms for solving compressed sensing problems.

Populärvetenskaplig Sammanfattning

Den snabba ökningen i beräkningskapacitet hos dagens datorer har de senaste åren banat väg för utveckling av en rad kraftfulla verktyg och metoder inom bildteknik, s.k. “computational photography”, inom vilka bildsensorer och optiska uppställningar kombineras med beräkningar för att skapa nya bildtillämpningar. Den visuella data som beräknas fram är oftast av högre dimensionalitet än vanliga bilder i 2D, dvs. resultatet är inte en bild som består av ett plan av bildpunkter i 2D utan en datamängd i 3D, 4D, 5D, ..., n D. Det grundläggande målet med dessa metoder är att ge användaren helt nya verktyg för att analysera och uppleva data med nya typer av interaktion- och visualiseringstekniker. Inom datorgrafik och datorseende, som är de drivande forskningsområdena, har det t.ex. introducerats tillämpningar såsom 3D-video (stereo), ljusfält (dvs. bilder och video där användaren i efterhand i 3D interaktivt kan ändra betraktningssvinkel, fokus eller zoom-nivå), displayer som tillåter 3D utan 3D-glasögon, metoder för mycket högupplöst skanning av miljöer och objekt, och metoder för mätning av materials optiska egenskaper för foto-realistisk visualisering av t.ex. produkter. Inom områden såsom radiologi och säkerhet hittar vi också bildtillämpningar, t.ex. röntgen, datortomografi eller magnetresonanskamera där beräkningar och sensorer kombineras för att skapa data som kan visualiseras i 3D, eller 4D om mätningen är utförd över tid. Ett centralt problem hos alla dessa tekniker är att de genererar mycket stora datamängder, ofta i storleksordningen hundratals GB eller till och med flera TB. En viktig forskningsfråga är därför att utveckla ny teori och praktiska metoder för effektiv kompression och lagring av högdimensionell visuell data. En lika viktig aspekt är att utveckla representationer och algoritmer som utöver effektiv lagring tillåter interaktiv databehandling, visualisering och uppspelning av data, t.ex. ljusfältsvideo.

Den här avhandlingen introducerar en rad nya representationer för inspelning, databehandling och lagring av högdimensionell visuell data. Den omfattar två huvudområden: effektiva representationer för kompression, databehandling och visualisering, samt effektiv mätning av högdimensionella visuella signaler baserat på s.k. compressive sensing. Avhandlingen och de ingående artiklarna introducerar en rad bidrag i form av teori, algoritmer och metoder inom båda dessa områden.

Det första huvudområdet i avhandlingen fokuserar på utveckling av en uppsättning datarepresentationer som med hjälp av maskininlärnings- och optimeringsmetoder anpassar sig till data. Detta möjliggör optimala representationer baserat på ett antal kriterier, t.ex. att representationen ska vara så kompakt (komprimerad) som möjligt och att de approximationsfel som introduceras vid rekonstruktion

av signalen är så små som möjligt. Dessa representationer bygger på s.k. glesa (på engelska “sparse”) basfunktioner. En uppsättning basfunktioner kallas för en ordbok (“dictionary”) och en signal, t.ex. en bild, video eller ett ljusfält, representeras som en kombination av basfunktioner från ordboken. Om representationen tillåter att signalen kan beskrivas med ett litet antal basfunktioner, motsvarande en mindre informationsmängd än originalsignalen, leder detta till att signalen komprimeras. I den teori och de tekniker som har utvecklats inom ramarna för den här avhandlingen optimeras basfunktionerna och ordboken utifrån den typ av data som ska representeras. Detta gör det möjligt att beskriva en signal med ett mycket litet antal basfunktioner, vilket leder till mycket effektiv komprimering och lagring. De utvecklade representationerna är specifikt designade så att alla centrala beräkningar kan ske parallellt per datapunkt. Genom att utnyttja kraften hos moderna GPUer är det därför möjligt att i realtid hantera och visualisera mycket minneskrävande visuella signaler.

Det andra området i avhandlingen tar sin startpunkt i att de representationer och basfunktioner som har utvecklats för kompression gör det möjligt att på ett mycket effektivt sätt applicera moderna mät- och samplingsmetoder, compressive sensing, på bilder, video, ljusfält och andra visuella signaler. Den teori compressive sensing bygger på visar att en signal kan mätas och rekonstrueras från ett mycket lågt antal mätpunkter om mätningen utförs i en bas, representation, där signalen som mäts är gles/sparse. Genom att utnyttja denna egenskap utvecklar avhandlingen teori för hur compressive sensing kan användas för mätning och rekonstruktion av visuella signaler oavsett dess dimensionalitet samt demonstrerar genom en rad olika exempel hur compressive sensing kan appliceras i praktiken.

Acknowledgments

I have been fortunate enough to work and collaborate with the most amazing and knowledgeable people I have ever met through my years as a PhD student. Therefore, I found writing this section of my thesis to be the most difficult one because putting into words my sincere gratitudes towards my supervisor, co-supervisor, colleagues, friends, and family is NP-hard. But I will try.

First and foremost, I would like to thank my supervisor, *Jonas Unger*, for his guidance, support, and patience throughout my years as a PhD student. Your enthusiasm for research always encouraged me to do more and more. What a journey this was! Hectic at times, but never ceasing to be fun. I cannot imagine being the academically grown person I am today without your guidance. And I will be forever grateful for the balance you provided between the supervision and the freedom to do research independently. If it wasn't for the academic freedom you provided, I wouldn't have discovered my favorite research topics. And if it wasn't for your guidance, I wouldn't have progressed as much in those topics. It has been a privilege working with you and I hope that our collaborations will continue. I would like to thank my co-supervisor, *Anders Ynnerman*. You have established a research environment with the highest standards and I thank you for giving me the chance to be a PhD student at the MIT division.

I would like to thank my colleagues at the computer graphics and image processing group. I am very grateful for being among you during my PhD studies. First, *Per Larsson*, the man behind all the cool research gadgets I used during my studies. Funny, kind, and intelligent, you have it all Per! *Gabriel Eilertsen*, the HDR guru in our group with the most amazing puns. Thank you *Saghi Hajisharif*, you simply are the best (my totally unbiased opinion). I am very grateful for all the good collaborations we had. Thank you *Tanaboon Tongbuasirilai* for helping me with the BRDF renderings included in this thesis and all the enjoyable discussions we had. Thank you *Apostolia Tsirikoglou*, our deep learning expert, for lighting up the office with your jokes. And thanks to the previous members of the group with whom I spent the majority of my PhD studies. Specifically, thank you *Joel Kronander* for all the memorable discussions we had as PhD students from compressed sensing to rendering and beyond! Thank you for the collaborations we had; it was a privilege and I truly miss you in the lab. And by the way, "spike ensemble" is the new "sub-identity". Thank you *Andrew Gardner*, also known as the coding guru, for our collaborations on the compression library. We miss you! And last but not least, thank you *Eva Skärblom* for arranging everything from the travel plans to my salary!

Thank you *Mohammad Emadi* for two years of long distance collaboration on our

work towards Paper F and Paper G. You are a brilliant mind! I also would like to thank *Christine Guillemot*, the head of SIROCCO team at INRIA, Rennes, France, who gave me the opportunity for a research visit. I truly enjoyed my time there and I am grateful for our collaboration that led to Paper E.

Thank you Saghi for making my life what I always wanted it to be. I cannot imagine going through the difficult times without the hope and happiness you brought. And thank you for your understanding during the stressful time of writing this thesis. I am incomplete without you, to the extent that even compressed sensing cannot recover me.

Ali Samini, my dear boora, I am so happy that I came to Sweden because otherwise I most probably wouldn't have met you. I had some of the best times in my life during our years as PhD students and you have always been there in my difficult times. Thank you *Alexander Bock*, my German brother, for all the good times we had together during my PhD studies. Thanks to my dearest friends in Iran: *Aryan*, *Javid*, *Kamran* (the 3 + 1 musketeers). Every time I went back to Iran and visited you guys, all the tiredness gathered during the year from the deadlines just vanished.

And last but certainly not least, I would like to thank my parents, *Shamsi* and *Davood*. Thank you for placing such a high value on education since my childhood. You have always pushed me to achieve more on every step of my life. I am eternally grateful for everything you have done for me.

List of Publications

The published work of the author that are included in this thesis is listed below in reverse chronological order.

- E. Miandji, S. Hajisharif, and J. Unger, “A Unified Framework for Compression and Compressed Sensing of Light Fields and Light Field Videos,” *ACM Transactions on Graphics*, Provisionally accepted
- 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, Sept 2018
- E. Miandji[†], M. Emadi[†], and J. Unger, “OMP-based DOA Estimation Performance Analysis,” *Digital Signal Processing*, vol. 79, pp. 57–65, Aug 2018, [†] equal contributor
- 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, [†] equal contributor
- E. Miandji and J. Unger, “On Nonlocal Image Completion Using an Ensemble of Dictionaries,” in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, Sept 2016, pp. 2519–2523
- E. Miandji, J. Kronander, and J. Unger, “Compressive Image Reconstruction in Reduced Union of Subspaces,” *Computer Graphics Forum*, vol. 34, no. 2, pp. 33–44, May 2015
- 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*. ACM, 2013, pp. 24:1–24:4

Other publications by the author that are relevant to this thesis but were not included are:

- G. Baravdish, E. Miandji, and J. Unger, “GPU Accelerated Sparse Representation of Light Fields,” in *14th International Conference on Computer Vision Theory and Applications (VISAPP 2019)*, submitted

- E. Miandji[†], M. Emadi[†], and J. Unger, “A Performance Guarantee for Orthogonal Matching Pursuit Using Mutual Coherence,” *Circuits, Systems, and Signal Processing*, vol. 37, no. 4, pp. 1562–1574, Apr 2018, [†] equal contributor
- 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, May 2015
- S. Mohseni, N. Zarei, E. Miandji, and G. Ardeshtir, “Facial Expression Recognition Using Facial Graph,” in *Workshop on Face and Facial Expression Recognition from Real World Videos (ICPR 2014)*. Cham: Springer International Publishing, 2014, pp. 58–66
- S. Hajisharif, J. Kronander, E. Miandji, and J. Unger, “Real-time Image Based Lighting with Streaming HDR-light Probe Sequences,” in *SIGRAD 2012: Interactive Visual Analysis of Data*. Linköping University Electronic Press, Nov 2012
- E. Miandji, J. Kronander, and J. Unger, “Geometry Independent Surface Light Fields for Real Time Rendering of Precomputed Global Illumination,” in *SIGRAD 2011*. Linköping University Electronic Press, 2011
- E. Miandji, M. H. Sargazi Moghadam, F. F. Samavati, and M. Emadi, “Real-time Multi-band Synthesis of Ocean Water With New Iterative Up-sampling Technique,” *The Visual Computer*, vol. 25, no. 5, pp. 697–705, May 2009

Contributions

In what follows, the main publications included in this thesis are listed, where a short description of each paper along with author’s contributions are presented.

Paper A: Learning Based Compression of Surface Light Fields for Real-time Rendering of Global Illumination Scenes

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*. ACM, 2013, pp. 24:1–24:4

This paper presents a method for real time photorealistic precomputed rendering of static scenes with arbitrarily complex materials and light sources. To achieve this, a training-based method for compression of Surface Light Fields (SLF) using an ensemble of orthogonal two-dimensional dictionaries is presented. We first generate an SLF for each object of a scene using the PBRT library [15]. Then, an ensemble is trained for each SLF in order to represent the SLF using sparse coefficients. To accelerate training and exploit the local similarities in each SLF, the authors use K-Means [16] for clustering prior to training an ensemble. Real time performance is achieved by local reconstruction of the compressed set of SLFs using the GPU. The author of this thesis was responsible for the design and implementation of the algorithm, as well as the majority of the written presentation.

Paper B: 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 Transactions on Graphics*, Provisionally accepted

This work is an extension of Paper A with several contributions. We propose a method for training n -dimensional (nD) dictionaries that enable enhanced sparsity compared to 2D and 1D variants used in the previous work. The proposed method admits sparse representation and compressed sensing of arbitrarily high dimensional datasets, including light fields (5D), light field videos (6D), bidirectional texture functions (7D) and spatially varying BRDFs (7D). We also observed that K-Means is not adequate for capturing self-similarities of complex high dimensional datasets.

Therefore, we present a pre-clustering method that is based on sparsity, rather than the ℓ_2 norm, while being resilient to noise and outliers. The ensemble of multidimensional dictionaries enables real-time rendering of large-scale light field videos. The author of this thesis was responsible for the design of the framework, and contributed to the majority of the implementation and written presentation.

Paper C: Compressive Image Reconstruction in Reduced Union of Subspaces

E. Miandji, J. Kronander, and J. Unger, “Compressive Image Reconstruction in Reduced Union of Subspaces,” *Computer Graphics Forum*, vol. 34, no. 2, pp. 33–44, May 2015

This paper presents a novel compressed sensing framework using an ensemble of trained 2D dictionaries. The framework was applied for compressed sensing of images, videos, and light fields. While the trained ensemble is 2D, the compressed sensing was applied to 1D patches using the construction of a Kronecker ensemble. Using point sampling as the sensing matrix, the framework is shown to be effective for accelerating photo realistic rendering, as well as light field imaging. The results show significant improvements over state of the art methods in graphics and image processing literature. The author was responsible for the design and implementation of the method, as well as the majority of its written presentation.

Paper D: On Nonlocal Image Completion Using an Ensemble of Dictionaries

E. Miandji and J. Unger, “On Nonlocal Image Completion Using an Ensemble of Dictionaries,” in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, Sept 2016, pp. 2519–2523

This paper presents a theoretical analysis on the compressed sensing framework introduced in Paper C. In particular, we derive a lower bound on the number of point samples and the probability of success for exact recovery of an image or a light field. The lower bound is dependent on the coherence of dictionaries in the ensemble. The theoretical results presented are useful for designing efficient dictionary ensembles for light field imaging. In Paper B, we reformulate these results for n D dictionary ensembles. The author was responsible for the design, implementation, and the written presentation of the method. The paper was presented by Jonas Unger at ICIP 2016, Phoenix, Arizona.

Paper E: Multi-Shot Single Sensor Light Field Camera Using a Color Coded Mask

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, Sept 2018

In this paper, a compressive light field imaging system based on the coded aperture design is presented. A color coded mask is placed between the aperture and the sensor. It is shown that the color coded mask is significantly more effective than the previously used monochrome masks [17]. Moreover, by micro movements of the mask and capturing multiple shots, one can incrementally improve the reconstruction quality. In addition, spatial sub-sampling is proposed to provide a trade off between the quality and speed of reconstruction. The proposed method achieves up to 4dB in PSNR over [17]. The majority of this work was done when the author was working as a visiting researcher at SIROCCO lab, a leading light field research group at INRIA, Rennes, France, led by Christine Guillemot. The author was responsible for the design and implementation of the method, and contributed to the majority of the written presentation.

Paper F: On Probability of Support Recovery for Orthogonal Matching Pursuit Using Mutual Coherence

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, [†] equal contributor

This paper presents a novel theoretical analysis of the Orthogonal Matching Pursuit (OMP), one of the most commonly used greedy sparse recovery algorithm for compressed sensing. We derive a lower bound for the probability of identifying the support of a sparse signal contaminated with additive Gaussian noise. Mutual coherence is used as a metric for the quality of a dictionary and we do not assume any structure for the dictionary, or the sensing matrix; hence, the theoretical results can be utilized in any compressed sensing framework, e.g. Paper B, Paper C, and Paper E. The new bound significantly outperforms the state of the art [18]. The author of this thesis and Mohammad Emadi equally contributed to this work. The project was carried out in collaboration with the Cornell University (and later on the University of Michigan).

Paper G: OMP-based DOA estimation performance analysis

E. Miandji[†], M. Emadi[†], and J. Unger, “OMP-based DOA Estimation Performance Analysis,” *Digital Signal Processing*, vol. 79, pp. 57–65, Aug 2018, [†] equal contributor

Presents an extension of the theoretical work in Paper F. We derive an upper bound for the Mean Square Error (MSE) of OMP given a dictionary, the signal sparsity, and the signal's noise properties. Using this upper bound, we derive new lower bounds for the probability of support recovery. Compared to Paper F, the new bounds are formulated using more "user-friendly" parameters such as the signal to noise ratio and the dynamic range. Furthermore, these new bounds shed light on the special properties of the bound derived in Paper F. In addition, here we consider the more general case of complex valued signals and dictionaries. Although the application considered in the paper is estimating the Direction of Arrival (DOA) for sensor arrays, the theoretical results presented are relevant to this thesis. The author of this thesis and Mohammad Emadi equally contributed to this work.

Contents

Abstract	v
Populärvetenskaplig Sammanfattning	vii
Acknowledgments	ix
List of Publications	xi
Contributions	xiii
1 Introduction	1
1.1 Sparse Representation	4
1.2 Compressed Sensing	9
1.3 Thesis outline	11
1.4 Aim and Scope	12
2 Preliminaries	15
2.1 Notations	15
2.2 Tensor Approximations	16
2.3 Sparse Signal Estimation	19
2.4 Dictionary Learning	22
2.5 Visual Data: The Curse/Blessing of Dimensionality	26
3 Learning-based Sparse Representation of Visual Data	29
3.1 Outline and Contributions	30
3.2 Multidimensional Dictionary Ensemble	30
3.2.1 Motivation	31
3.2.2 Training	34
3.2.3 Testing	37
3.3 Pre-clustering	39
3.3.1 Motivation	39
3.3.2 Aggregate Multidimensional Dictionary Ensemble	41
3.4 Summary and Future Work	43
4 Compression of Visual Data	45
4.1 Outline and Contributions	45
4.2 Precomputed Photorealistic Rendering	46
4.2.1 Background and Motivation	46
4.2.2 Overview	48
4.2.3 Data Generation	48

4.2.4	Compression	48
4.2.5	Real Time Rendering	49
4.2.6	Results	51
4.3	Light Field and Light Field Video	51
4.4	Large-scale Appearance Data	55
4.5	Summary and Future Work	57
5	Compressed Sensing	59
5.1	Outline and Contributions	60
5.2	Definitions	61
5.3	Problem Formulation	62
5.4	The Measurement Matrix	63
5.5	Universal Conditions for Exact Recovery	64
5.5.1	Uncertainty Principle and Coherence	64
5.5.2	Spark	67
5.5.3	Exact Recovery Coefficient	67
5.5.4	Restricted Isometry Constant	68
5.6	Orthogonal Matching Pursuit (OMP)	70
5.7	Theoretical Analysis of OMP Using Mutual Coherence	72
5.7.1	The Effect of Noise on OMP	73
5.7.2	Prior Art and Motivation	73
5.7.3	A Novel Analysis Based on the Concentration of Measure	75
5.7.4	Numerical Results	78
5.7.5	User-friendly Bounds	78
5.8	Uniqueness Under a Dictionary Ensemble	79
5.8.1	Problem Formulation	80
5.8.2	Uniqueness Definition	81
5.8.3	Uniqueness with a Spike Ensemble	82
5.8.4	Numerical Results	85
5.9	Summary and Future Work	85
6	Compressed Sensing in Graphics and Imaging	89
6.1	Outline and Contributions	91
6.2	Light Field Imaging	91
6.2.1	Prior Art	91
6.2.2	Multi-shot Single-sensor Light Field Camera	94
6.2.3	Implementation and Results	95
6.3	Accelerating Photorealistic Rendering	98
6.3.1	Prior Art	98
6.3.2	Compressive Rendering	99
6.3.3	Compressive Rendering Using MDE	99
6.3.4	Implementation and Results	100

6.4	Compressed Sensing of Visual Data	103
6.4.1	Light Field and Light Field Video Completion	103
6.4.2	Compressive BRDF capture	104
6.5	Summary and Future Work	108
7	Concluding Remarks and Outlook	111
7.1	Theoretical Frontiers	112
7.2	Model Improvements	113
7.3	Applications	114
7.4	Final Remarks	116
	Bibliography	119
	Publications	161
	Paper A	163
	Paper B	171
	Paper C	189
	Paper D	205
	Paper F	213
	Paper E	221
	Paper G	229

Introduction

In the past decade, massive amounts of high dimensional data in various fields of science and engineering have been produced. Not only the amount of data created nowadays is beyond the current processing and storage capabilities, the speed at which these datasets are being produced is increasing. While the capturing, storage, and the processing of large-scale datasets pose various challenges, high dimensional datasets have presented many research opportunities in areas such as visual data processing, bioinformatics, web analytics, biomedical imaging, and many more. Large-scale high dimensional data has moved scientific discoveries to a new paradigm, what is known as the fourth paradigm of discovery [19].

The ongoing advances in imaging techniques have introduced a range of new large-scale visual data such as light field images [20] and video [21], measured BRDFs [22] and Spatially Varying BRDFs (SVBRDF) [23], multispectral images [24], Bidirectional Texture Functions (BTF) [25], and Magnetic Resonance Imaging (MRI) [26]. A common feature of these datasets is high dimensionality, see Fig. 1.1. For instance, a light field is a 5D function defined as $l(r, t, u, v, \lambda)$, where (r, t) describes the spatial domain, (u, v) parametrizes the angular domain, and λ represents wavelength. A BRDF, can be parametrized as a 4D, 5D, or 6D object depending on the way we define the dependence on wavelengths. Recent advances in imaging technologies enable capturing of these datasets with a high resolution along each dimension. Moreover, existing means for capturing such datasets can be extended to accommodate more information, e.g. for acquiring multispectral light field videos.

The various types of high dimensional data described above have found novel and

Figure 1.1: Examples of high dimensional datasets used in computer graphics.

Figure 1.2: A pipeline for utilizing high dimensional visual data in graphics

important applications in many areas within computer graphics and vision. For instance, light fields have been widely used for designing glasses-free 3D displays [27, 28, 29] and photo-realistic real time rendering, see Paper A and [30, 31]. BTFs have been used in a wide range of applications such as estimating BRDFs [32], theoretical analysis of cast shadows [33], real-time rendering [34, 35], and geometry estimation [36]. Moreover, measured BRDF datasets such as [37] have enabled more than a decade of research in deriving new analytical models [38, 39, 40] and evaluating existing ones [41], as well as efficient photo-realistic rendering.

Given the discussion above, it is clear that there are several stages for incorporating high dimensional visual data in graphics, each comprising of numerous research directions (see Fig. 1.2). The first component is to derive means for efficient capturing of multidimensional visual data with high fidelity for different applications. Typically, this involves the use of multiple cameras and sensors. For portable measurement devices, the amount of data produced is often orders of magnitude higher than the capabilities of storage devices (I/O speed, space, etc.). Moreover, real-time compression of large-scale streaming data is either infeasible or very

costly. However, it is possible to *directly* measure the compressed data, which will bypass two costly stages: namely the storage of raw data followed by compression. This is the fundamental idea behind *compressed sensing* [42], a relatively new field in applied mathematics and signal processing. Section 1.2 presents a brief description of the contributions of this thesis in the field of compressed sensing. These contributions advance the field in two fundamental areas: theoretical and empirical.

Another important aspect regarding multidimensional visual data is to find alternative representations of the data that facilitate storage, processing, and fast reconstruction. In order to derive suitable *basis* functions for efficient representation of the data, it is required to have accurate data models. An important reason for finding transformations of the data is that the original domain of the data is too complicated to be analyzed, stored, or rendered. With the ever growing storage cost of these datasets and the limitations of processing power and memory, such representations have been a fundamental component in many applications since the early days of computer graphics. For instance, Spherical Harmonics have been used for BRDF inference [43], and representation of radiance transfer functions [44]. Fourier domain has been used for theoretical analysis of light transport [45, 46] and various other applications [47, 48, 49]. Furthermore, wavelets have also been shown to be an important tool in graphics [50, 51, 52, 53].

Throughout this thesis, we use the word *dictionary* to describe a set of basis functions. Just like a sentence or a speech is a linear combination of words from a dictionary, multidimensional visual data can be formed using a linear combination of *atoms*. An atom is a basis function or a common feature in a dataset. A collection of atoms forms a dictionary. Dictionaries can be divided into two major groups [54]: *analytical* and *learning-based*. Analytical dictionaries are based on a mathematical model of the data, where an equation or a series of equations are used to effectively represent the data. The Fourier basis, wavelets, curvelets [55], and shearlets [56] are a few examples of dictionaries in this group. On the other hand, machine learning techniques can be used on a large set of examples to obtain a learning-based dictionary. While learning-based dictionaries have been shown to produce better results due to their adaptivity, they are often computationally expensive and pose various challenges for large signal sizes.

Of particular interest in many signal and image processing applications is *sparse* representations, where a signal can be modeled by linear combination of a small subset of atoms in the dictionary. Since a signal is modeled using a few scalars, compression is a direct byproduct of sparse representations. Visual data such as natural images and light fields admit sparse representations if a suitable dictionary is given or trained. The quality of representation, i.e. the sparsity and the amount of error introduced by the model, are directly related to the dictionary. Indeed,

the more sparse the representation is, the higher the compression ratio becomes. This thesis makes several contributions in this direction. In particular, we propose algorithms that enable highly sparse representations of multidimensional visual data, as well as various applications that utilize this model for efficient compression of large scale datasets in graphics. The enhanced sparsity of the representation is also a key component for compressed sensing. The more sparse a signal is, the less samples are required to *exactly* reconstruct the signal. We briefly describe and formulate sparse representations in Section 1.1, followed by the contributions of this thesis on the topic.

1.1 Sparse Representation

Data models have had a central contribution during the past few decades in image processing and computer graphics [57]. Fundamental problems such as sampling, denoising, super-resolution, and classification cannot be tackled effectively without prior assumptions about the signal model. Indeed natural visual data such as images and light fields contain structures and features that can be modeled effectively to facilitate reconstruction, sampling, or even synthesizing new examples of [58, 59]. If such structures did not exist, one could hope for creating an image of a cat by sampling from a uniform distribution – a very unlikely outcome. Sparse representation is a signal model that has been successfully applied in many image processing applications, ranging from solving inverse problems [60, 61, 62] to image classification [63, 64].

We first present an illustrative definition of sparse representation in a two dimensional space, followed by a general formulation of the sparse representation problem. Assume that a set of points (discrete signals) in a two dimensional space are given, see Fig. 1.3a. Indeed we need two scalars to describe the location of each point. If we rotate and translate the coordinate axes, see Fig. 1.3b, we obtain a new coordinate system, for which the red, green, and blue points can be represented with one scalar instead of two. We have now achieved a sparse representation for three of the points in a new orthogonal coordinate system. In other words, we have reduced the amount of information needed to represent the points. It can be noted that the blue point is not exactly sparse since it does not completely lie on one of axes. This is a common outcome in practice since a large family of signals are not exactly sparse. However, if a coordinate value is close to zero, we can assume sparsity, albeit at cost of introducing error in the representation.

In order to construct a coordinate system that admits a sparse representation for the black point, we can add a new coordinate axis, as shown in Fig. 1.3c. This might seem counter-intuitive since by adding a coordinate axis we have increased the amount of information needed for representing the points. However, the red,

Figure 1.3: An Illustration of (a) a set of points in a 2D space and their sparse representation using (b) an orthogonal dictionary, (c) an overcomplete dictionary, and (d) an ensemble of orthogonal dictionaries.

green, and blue points are already sparse. As a result, we now only need one scalar to represent each of the four points. Moreover, in practice, we have far more points than coordinate axes. The problem of finding the smallest set of coordinate axes that produces the most sparse representation for all the points is known as *dictionary learning*. From the discussion above, it can be deduced that each coordinate axis is an atom in a dictionary (coordinate system). Note that by adding the third atom in Fig. 1.3c, we lost the orthogonality of the dictionary. An alternative to adding a new atom is to construct an additional orthogonal dictionary, as shown in Fig. 1.3d. The majority of the work presented in this thesis uses a set of orthogonal dictionaries for sparse representation, which we refer to as an *ensemble of dictionaries*.

We now formulate the sparse representation of discrete signals in $\mathbb{R}^m$. Let the vector $\mathbf{x} \in \mathbb{R}^m$ be a discrete signal, e.g. a vectorized image. The linear system $\mathbf{D}\mathbf{s} = \mathbf{x}$ models the signal as a linear combination of k atoms (coordinate axes) that are arranged as the columns of a dictionary $\mathbf{D} \in \mathbb{R}^{m \times k}$ using the weights in

Figure 1.4: Examples of a compressible signal $\mathbf{s} \in \mathbb{R}^{100}$ for $C = 1$ and different values of q . The values of $\mathbf{s}$ are assumed to be sorted in decreasing order based on their magnitude.

Figure 1.5: Sparse representation of an image. The insets show the image quality when we use a percentage of nonzero coefficients that are obtained with two types of dictionaries: DCT (an analytical dictionary used in JPEG) and AMDE (a learning based multidimensional dictionary ensemble introduced in Paper B).

the vector $\mathbf{s} \in \mathbb{R}^k$. If $k > m$, then the dictionary is called *overcomplete*, as shown in Fig. 1.3c, and the linear system has infinitely many solutions for $\mathbf{s}$ when $\mathbf{D}$ is full-rank. However, if we limit the space of solutions for $\mathbf{s}$ to those with at most $\tau \leq m$ nonzero values, it is possible to obtain a unique solution under certain conditions, see e.g. [65]. This is indeed under the assumption of signal sparsity. In some cases, specifically for visual data, the signal is not sparse but *compressible*, which we define below:

Definition 1.1 (compressible signal). A signal $\mathbf{x} \in \mathbb{R}^m$ is called compressible under a dictionary $\mathbf{D} \in \mathbb{R}^{m \times k}$ if $\mathbf{x} = \mathbf{D}\mathbf{s}$ and the sorted magnitude of elements in $\mathbf{s}$ obey the power law, i.e. $|\mathbf{s}_j| \leq Cj^{-q}$, $j \in \{1, 2, \dots, k\}$, where C is a constant.

Examples of signals obeying the power law are given in Fig. 1.4. It is well known that visual data such as images, videos, light fields, etc., are compressible rather than sparse. In other words, the vector $\mathbf{s}$ is unlikely to have elements that are exactly zero; however, the majority of elements are close to zero. Hence, a compressible signal can be converted to a sparse signal by nullifying the smallest elements of the vector $\mathbf{s}$. This process will indeed introduce error in the representation. Nevertheless, since the nullified elements are small in magnitude, the error is typically negligible. In Fig. 1.5, we show the effect of nullifying small coefficients of a compressible signal (an image in this case) on the image quality. In particular, the effect of increasing τ/m , i.e. the ratio of nonzero coefficients to the signal length, on image quality is shown. Indeed having more nonzero coefficients leads to a higher image quality while increasing the storage complexity. Moreover, having more nonzero coefficients requires more samples when we would like to sample a sparse signal, see Section 1.2 for more details.

Sparse representation can be defined as the problem of finding the most sparse set of coefficients for a given signal with minimal error. When an analytical overcomplete dictionary is given, sparse representation amounts to solving a least squares problem with a constraint on the sparsity or the representation error. However, we also need to find a suitable dictionary that enables sparse representation. As mentioned earlier in this chapter, one approach is to construct a dictionary from a set of examples using machine learning, which will be described in more detail in Section 2.4. Therefore, to obtain the most sparse representation of a set of signals, we have two unknowns: 1. the dictionary and 2. the set of sparse coefficients. Estimating each of these entities requires an estimate of the other. Therefore, it is a common practice to solve this problem by alternating between the estimation of the coefficients and the dictionary, which is a fundamental problem in sparse representation. Perhaps the most eloquent description of this fundamental problem is given by A. M. Bruckstein in the Foreword of the first book on the topic [66]:

“The field of sparse representations, that recently underwent a Big-Bang-like expansion, explicitly deals with the Yin-Yang interplay between the parsimony of descriptions and the “language” or “dictionary” used in them, and it became an extremely exciting area of investigation.”

This thesis makes contributions on finding suitable models for sparse representation of multidimensional visual data that promote sparsity, while reducing the representation error. Sparse representation of multidimensional visual data is the topic of

Chapter 3, where we describe a highly effective dictionary learning method that enables very sparse representations with minimal error. The proposed method is utilized in a variety of applications, discussed in Chapter 4, including compression and real time photorealistic rendering. In the remainder of this section, a brief description of these applications is presented.

Indeed a direct consequence of sparse representation is compression since we only need to store the nonzero coefficients of a sparse signal. This has been a common practice for compression. For instance, the JPEG standard for compressing images uses an analytical dictionary, namely the Discrete Cosine Transform (DCT), to obtain and compress a set of sparse coefficients. Similarly, JPEG2000 [67] uses CDF 9/7 wavelets [68]. Recent video compression methods such as HEVC (H.265) also use a variant of DCT for compression. While analytical dictionaries are fast to evaluate, learning-based dictionaries significantly outperform them in terms of reconstruction quality and sparsity of coefficients. In this direction, Paper A presents a method for compression of Surface Light Field (SLF) datasets generated using a photo-realistic renderer. The proposed algorithm relies on a trained ensemble of orthogonal dictionaries that operates on the rows and columns of each Hemispherical Radiance Distribution Function (HRDF), independently. An HRDF contains the outgoing radiance along multiple directions at a single point on the surface of an object. The compression method admits real-time reconstruction using the Graphics Processing Unit (GPU). As a result, real-time photorealistic rendering of static scenes with highly complex materials and light sources is achieved.

As described previously, visual datasets in graphics are often multidimensional. For instance, images are 2D objects while a light field video is a 6D object. An efficient dictionary for sparse representation should accommodate datasets with different dimensionality. In Paper B, a dictionary training method is presented that computes an ensemble of n D orthonormal dictionaries. Moreover, a novel pre-clustering method is introduced that improves the quality of the learned ensemble while substantially reducing the computational complexity. The proposed method can be utilized for the sparse representation, and hence the compression, of any discrete dataset in graphics and visualization, regardless of dimensionality. Moreover, the sparse representation obtained by the dictionary ensemble enables real time reconstruction of large-scale datasets. For instance, we demonstrate real time rendering of high resolution light field videos on a consumer level GPU. The independence of the dictionaries on dimensionality admits the application of our method for a wide range of datasets including videos, light fields, light field videos, BTFs, measured BRDFs, SVBRDFs, etc.

1.2 Compressed Sensing

The Shannon-Nyquist theorem for sampling band-limited continuous-time signals [69, 70] formed a strong foundation for decades of innovation in designing new sensing systems. The theorem states that any function with no frequencies higher than f can be exactly recovered with equally spaced samples at a rate larger than $2f$, known as the Nyquist rate. In many applications, despite the rapid growth in computational power, designing systems that operate at the Nyquist rate is challenging [71]. One solution is to sample the signal densely and use compression with the help of sparse representations, as was discussed in Section 1.1. Although computationally expensive, this approach is widely used in many sensing systems; for instance, digital cameras for images, videos, and light fields use dense sampling followed by compression. However, sampling a signal densely and discarding redundant information through compression is a wasteful process. Therefore, an interesting question in this regard is: *Can we directly measure the compressed signal?* Compressed sensing addresses this question by utilizing the strong foundation of sparse representations. In essence, compressed sensing can be defined as the “*the art of sampling sparse signals*”.

Compressed sensing was first introduced in applied mathematics for solving underdetermined systems of linear equations and was quickly adopted by the signal processing and information theory communities to establish a completely different perspective on the sampling problem. Let $\mathbf{D} \in \mathbb{R}^{m \times k}$ be a dictionary and $\Phi \in \mathbb{R}^{s \times m}$ a linear sampling operator, with s being the number of samples. The operator Φ is typically called a *sensing matrix* or a *measurement matrix*. The formal definition of a sensing matrix will be given in Section 5.4. For now, we assume that Φ is a mapping from $\mathbb{R}^m$ to $\mathbb{R}^s$, where $s < m$. Using a linear measurement model, a signal $\mathbf{x} \in \mathbb{R}^m$ is sampled using Φ by evaluating $\mathbf{y} = \Phi\mathbf{x} + \mathbf{w}$, where $\mathbf{w}$ is the measurement noise, often assumed to be the white Gaussian noise. The signal $\mathbf{x}$ is assumed to be τ -sparse in the dictionary $\mathbf{D}$, i.e. we have $\mathbf{x} = \mathbf{D}\mathbf{s}$, where $\|\mathbf{s}\|_0 \leq \tau$, and the function $\|\cdot\|_0$ counts the number of nonzero values in a vector. In this setup, reconstructing the sparse signal from the measurements $\mathbf{y}$ involves solving the following optimization problem

$$\min_{\hat{\mathbf{s}}} \|\hat{\mathbf{s}}\|_0 \quad \text{s.t.} \quad \|\mathbf{y} - \Phi\mathbf{D}\hat{\mathbf{s}}\|_2^2 \leq \epsilon, \quad (1.1)$$

and then computing the signal estimate as $\hat{\mathbf{x}} = \mathbf{D}\hat{\mathbf{s}}$; the constant ϵ is related to the noise power. Since (1.1) is not convex, it is a common practice to solve the following problem instead

$$\min_{\hat{\mathbf{s}}} \|\hat{\mathbf{s}}\|_1 \quad \text{s.t.} \quad \|\mathbf{y} - \Phi\mathbf{D}\hat{\mathbf{s}}\|_2^2 \leq \epsilon. \quad (1.2)$$

However, a solution of (1.2) is not necessarily a solution of (1.1). Early work in compressed sensing derive necessary and sufficient conditions for the equivalence

of (1.1) and (1.2), see e.g. [72, 73, 74]. Of particular interest in designing sensing systems is the use of random sampling matrices, Φ , due to the simplicity of implementation and well-studied theoretical properties. Seminal work in the field [75, 76, 77, 78] show that a signal of length m with at most $\tau \leq m$ nonzero elements can be recovered with overwhelming probability using Gaussian or Bernoulli sensing matrices, provided that $s \geq C\tau \ln(m/\tau)$, where C is a universal constant. The result is significant since the number of samples is linearly dependent on sparsity, while being logarithmically influenced by the signal length. Therefore, compressed sensing can guarantee the exact recovery of sparse signals with vastly reduced number of measurements compared to the Nyquist rate.

Research in compressed sensing can be divided into two categories: theoretical and empirical. Theoretical research addresses fundamental problems such as the conditions for exact recovery of sparse signals from a few samples. In this regard, there exists two research directions. Universal results consider random measurement matrices and derive bounds that hold for every sparse signal. Moreover, there exists algorithm-specific theoretical analysis that derive convergence conditions for algorithms that solve (1.1) or (1.2) without any assumptions on the sensing matrix, see e.g. [79, 80, 81, 82]. Empirical research, on the other hand, apply the sparse sensing model described above for designing effective sensing systems. For instance, compressed sensing has been used for designing a digital camera with a single pixel [83], light field imaging [17, 84, 85], light transport sensing [86], Magnetic Resonance Imaging (MRI) [26, 87, 88], sensor networks [89, 90], and antenna arrays [91, 92, 93].

This thesis makes contributions in both theoretical and empirical aspects of compressed sensing, where the former is the topic of Chapter 5, and the latter is discussed in Chapter 6. The methods introduced herein show the applicability of compressed sensing in various subjects in graphics, while also presenting a theoretical analysis of the proposed algorithms. In this regard, Paper C presents a compressed sensing framework for recovering images, videos, and light fields from a small number of point samples. The framework utilizes an ensemble of orthogonal dictionaries trained on a collection of 2D small patches from various datasets. Moreover, it is shown that the method can be used for the acceleration of photorealistic rendering techniques without a noticeable degradation in image quality.

In Paper D, we perform a theoretical analysis of the framework presented in Paper C. This novel analysis derives uniqueness conditions for the solution of (1.1) when the signal is sparse in an ensemble of 2D orthogonal dictionaries. In other words, we show the required conditions for *exact* recovery of an image, light field, or any type of visual data from what appears to be a highly insufficient number of samples. The main result is a lower bound on the required number of samples

and a lower bound on the probability of exact recovery. These theoretical results are reformulated in Paper B, where we consider an ensemble of n D orthogonal dictionaries. The theoretical analysis provides insight into training more efficient multidimensional dictionaries, as well as designing effective capturing devices for light fields, BRDFs, etc. Additionally, in Paper E we propose a new light field camera design based on compressed sensing. By placing a random color coded mask in front the sensor of a consumer level digital camera, high quality and high resolution light fields can be captured.

As mentioned earlier, theoretical results on deriving optimality conditions for sparse recovery algorithms play an important role in compressed sensing. Understanding the behaviour of a sparse recovery algorithm with respect to the properties of the input signal, as well as the dictionary, greatly improves the design of novel sensing systems. For instance, in designing a compressive light field camera, parameters such as Signal to Noise Ratio (SNR), Dynamic Range (DR), and the properties of the sensing matrix (which is directly coupled with the design of the camera), play an important role in efficiency and flexibility of the system. In this direction, Paper F presents a theoretical analysis of Orthogonal Matching Pursuit (OMP), a greedy algorithm that is widely used for solving (1.1). We derive a lower bound for the probability of correctly identifying the location of nonzero entries (known as *support*) in a sparse vector. Unlike previous work, this new bound takes into account signal parameters, which as described earlier are important in practical applications. In Paper G, we extend these results by deriving an upper bound on the error of the estimated sparse vector (i.e. the result of solving (1.1)). Moreover, by combining the probability and error bounds, we derive new “user-friendly” bounds for the probability of successful support recovery. These new bounds shed light on the effect of various parameters for compressed sensing using OMP.

1.3 Thesis outline

The thesis is divided into two parts. The first part introduces background theory and gives an overview of the contributions presented in this thesis. The second part is a compilation of seven selected publications that provide more detailed descriptions of the research leading up to this thesis.

The first part of this thesis is divided into six chapters. As the title of this thesis suggests, our main focus is on compression and compressed sensing of visual data by the means of effective methods for sparse representations. In Chapter 2, important topics that lay the foundation of this thesis will be presented. Specifically, we discuss multi-linear algebra (utilized in Paper B), algorithms for recovery of sparse signals (used in all the included papers except for Paper A), dictionary learning (utilized in Paper A, Paper B, Paper C, Paper D, and Paper E), and different types

of visual data, as well as the challenges and opportunities that are associated with them. Chapter 3 will present novel techniques for efficient sparse representation of multidimensional visual data, followed by a number of applications in computer graphics that utilize these techniques for compression in Chapter 4. In Chapter 5 we revisit compressed sensing, which was briefly described in Section 1.2, and present fundamental concepts regarding the theory of sampling sparse signals. Then, the theoretical contributions of this thesis on compressed sensing will be presented. In Chapter 6, we discuss the applications of compressed sensing in graphics and imaging. A number of contributions such as light field imaging, photorealistic rendering, light field video sensing, and efficient BRDF capturing will be presented.

Each chapter will present the main contributions of this thesis on the topic, where we first present some background information, motivations, and how the contributions of the author address the limitations of current methods. We conclude each chapter with a short summary and a discussion of possible venues for future work on the topic. Finally, in Chapter 7 we summarize the materials presented in thesis and provide an outlook on the future of the field of sparse representations, in connection to compression and compressed sensing, for efficient acquisition, storage, processing, and rendering of visual data. We pose several research questions that will hopefully provide new directions for future research.

1.4 Aim and Scope

The aim of the research conducted by the author and presented in this thesis has been to derive methods and frameworks that are applicable to many different datasets utilized in computer graphics and image processing. While the main focus of the empirical results presented here is on light fields and light field videos, the compression and compressed sensing methods introduced in this thesis are applicable to BRDF, SVBRDF, BTF, light transport data, and hyper-spectral data, as well as new types of high dimensional datasets that are to appear in the future. The theoretical results of Paper D, Paper F, and Paper G have even a wider scope in terms of applicability in different areas of research on compressed sensing. Indeed any compressed sensing framework based on an overcomplete dictionary or an ensemble of dictionaries can utilize these results. More importantly, the theoretical results presented here can be used to guide the design of new sensing systems for efficient capturing of multidimensional visual data.

As mentioned earlier, a substantial part of the thesis discusses sparse representations for compression of visual data. However, transform coding by the means of sparse representations is typically a small part of a full compression pipeline. For instance, the JPEG standard uses Huffman coding [94] to further compress the sparse coefficients obtained from a DCT dictionary. Moreover, the HEVC codec uses a

series of advanced coding algorithms in different stages to reduce the redundancy of data in time domain. Therefore, although the term “compression” is used in this thesis, we only address the problem of finding the most sparse representation of visual data, and the coding of sparse coefficient is out of the scope of the thesis. However, it should be noted that any type of entropy coding technique can be implemented on top of algorithms we present here.

Preliminaries

In this Chapter, important topics that are utilized throughout this thesis will be discussed. We start by describing the mathematical notations used throughout this thesis in Section 2.1. Tensor algebra and methods for tensor approximations are presented in Section 2.2. In particular, we present simple operations on tensors and introduce Higher Order SVD (HOSVD), which has been widely used in computer graphics and image processing applications. Next, the sparse signal estimation is formulated in Section 2.3. Two major approaches for addressing this problem, i.e. greedy and convex relaxation methods, are discussed. Additionally, an extensive literature review of existing methods is presented. We also discuss some of the recent advances in solving the high dimensional variant of the problem. In Section 2.4, we formulate and discuss various methods for dictionary learning. We start by presenting the most commonly used dictionary learning method (K-SVD). Then a comprehensive review of existing methods is presented. Moreover, the high dimensional dictionary learning problem is formulated and recent methods for solving it are discussed. Finally, a short description of common types of visual data in computer graphics are presented in Section 2.5. Additionally, we discuss the challenges and opportunities associated with the capturing, storage, and the rendering of these datasets.

2.1 Notations

Throughout this thesis, the following notational convention is used. Vectors and matrices are denoted by boldface lower-case (**a**) and bold-face upper-case letters

$(\mathbf{A})$, respectively. Tensors are denoted by boldface calligraphic letters, e.g. $\mathcal{A}$. A finite set of objects is indexed by superscripts, e.g. $\{\mathbf{A}^{(i)}\}_{i=1}^N$ or $\{\mathcal{A}^{(i)}\}_{i=1}^N$. In some cases, for convenience, we may use multiple indices for these sets, e.g. $\{\mathcal{A}^{(i,j)}\}_{i=1,j=1}^{N,M}$. Individual elements of $\mathbf{a}$, $\mathbf{A}$, and $\mathcal{A}$ are denoted $\mathbf{a}_i$, $\mathbf{A}_{i_1, i_2}$, $\mathcal{A}_{i_1, \dots, i_N}$, respectively. The i th column and row of $\mathbf{A}$ are denoted $\mathbf{A}_i$ and $\mathbf{A}_{i,:}$, respectively. Similarly, the j^{th} tensor fiber is denoted by $\mathcal{A}_{i_1, i_2, \dots, i_{j-1}, i_{j+1}, \dots, i_N}$. The function $\text{vec}(\mathcal{A})$ performs vectorization of its argument; e.g. the elements of a tensor along rows, columns, depth, etc. are arranged in a vector. Given an index set, I , the sub-matrix $\mathbf{A}_I$ is formed from columns of $\mathbf{A}$ indexed by I . The N-mode product of a tensor with a matrix is denoted by $\times_N$, e.g. $\mathcal{X} \times_N \mathbf{B}$. The Kronecker product of two matrices is denoted $\mathbf{A} \otimes \mathbf{B}$ and the outer product of two vectors is denoted $\mathbf{a} \circ \mathbf{b}$.

The ℓ_p norm of a vector $\mathbf{s}$, for $1 \leq p \leq \infty$, is denoted by $\|\mathbf{s}\|_p$. The induced operator norm for a matrix is denoted $\|\mathbf{A}\|_p$. By definition we have $\|\mathbf{A}\|_2 = \lambda_{\max}(\mathbf{A})$, where $\lambda_{\max}(\cdot)$ denotes the largest singular value of a matrix (in absolute value). Frobenius norm of a matrix is denoted $\|\mathbf{A}\|_F$. The ℓ_0 pseudo-norm of a vector, denoted $\|\mathbf{s}\|_0$, defines the number of non-zero elements. The location of non-zero elements in $\mathbf{s}$, also known as the support set, is denoted $\text{supp}(\mathbf{s})$. Consequently, $\|\mathbf{s}\|_0 = |\text{supp}(\mathbf{s})|$, where $|\cdot|$ denotes set cardinality. Occasionally, the exponential function, e^x , is denoted $\exp(x)$. Moore-Penrose inverse of a matrix is denoted $\mathbf{A}^\dagger$. Probability of an event B is denoted $\Pr\{B\}$ and the expected value of a random variable x is shown as $\mathbb{E}\{x\}$. For defining a variable, we use the symbol $\triangleq$; for instance, $\mathbf{C} \triangleq \mathbf{A} \otimes \mathbf{B}$.

2.2 Tensor Approximations

The term “*tensor*” is indeed an ambiguous term with various definitions in different fields of science and engineering. The main definition of a tensor comes from the field of differential geometry. A tensor is an invariant geometric object that does not depend on the choice of the local coordinate system on a manifold. In particular, a tensor of type (p, q) of rank $p+q$ is an object defined in each coordinate system $\{x^{(i)}\}_{i=1}^n$ by the set of numbers $\mathcal{T}_{j_1, \dots, j_q}^{i_1, \dots, i_p}$ such that a coordinate substitution $\{x^{(i)}\}_{i=1}^n \rightarrow \{y^{(i')}\}_{i'=1}^n$ is according to the law [95]

$$\mathcal{T}_{j'_1, \dots, j'_q}^{i'_1, \dots, i'_p} = \frac{\partial y^{(i'_1)}}{\partial x^{(i_1)}} \cdots \frac{\partial y^{(i'_p)}}{\partial x^{(i_p)}} \frac{\partial x^{(j_1)}}{\partial y^{(j'_1)}} \cdots \frac{\partial x^{(j_q)}}{\partial y^{(j'_q)}} \mathcal{T}_{j_1, \dots, j_q}^{i_1, \dots, i_p}, \quad (2.1)$$

where we have used the Einstein summation convention. Tensor algebra was the main mathematical tool in deriving the theory of general relativity. In the fields of computer graphics and image processing, the term tensor is often used to define

higher-order matrices, also known as multidimensional arrays. This definition does not necessarily consider the transformation law in (2.1), and is only defined in $\mathbb{R}^n$. We also follow this convention, i.e. the term “tensor” in this thesis refers to a multidimensional array of scalars.

Tensors are omnipresent in various applications such as computer graphics [96, 97, 98, 99, 100], imaging techniques [101, 102, 103, 104], image processing and computer vision [105, 106, 107, 108], as well as scientific visualization [109, 110, 111]. While tensors are merely an extension of matrices to dimensions larger than two, many of the tools used in matrix analysis are not applicable to tensors. In what follows, a brief description of a few tensor operations, e.g. unfolding and decomposition, will be presented. Note that the concepts relevant to this thesis will be covered and for more details the reader is referred to a comprehensive review article by Kolda and Bader [112] and a book on the topic [113].

Let $\mathcal{X} \in \mathbb{R}^{m_1 \times m_2 \times \dots \times m_n}$ be a real-valued n -dimensional (n D) tensor, also referred to as an n -way or n -mode tensor. The values along a certain mode of a tensor is called a fiber. For instance, second mode fibers of a 3D tensor $\mathcal{Z} \in \mathbb{R}^{m_1 \times m_2 \times m_3}$ are obtained from $\mathcal{Z}_{i_1, :, i_3}$ for different values of i_1 and i_3 .

The norm of a tensor $\mathcal{X}$ is calculated as

$$\|\mathcal{X}\| = \sqrt{\sum_{i_1=1}^{m_1} \sum_{i_2=1}^{m_2} \dots \sum_{i_n=1}^{m_n} \mathcal{X}_{i_1, i_2, \dots, i_n}^2}. \quad (2.2)$$

Multiplication of a tensor with a matrix along mode N is called the N -mode product and is denoted by the symbol $\times_N$. Given a tensor $\mathcal{X} \in \mathbb{R}^{m_1 \times m_2 \times \dots \times m_n}$ and a matrix $\mathbf{U} \in \mathbb{R}^{J \times m_N}$, the result of the N -mode product is a tensor of dimension $m_1 \times \dots \times m_{N-1} \times J \times m_{N+1} \times \dots \times m_n$ with elements

$$(\mathcal{X} \times_N \mathbf{U})_{i_1, \dots, i_{N-1}, j, i_{N+1}, \dots, i_n} = \sum_{i_N=1}^{m_N} \mathcal{X}_{i_1, i_2, \dots, i_{N-1}, i_N, i_{N+1}, \dots, i_n} \mathbf{U}_{j, i_N}. \quad (2.3)$$

Another useful operation on tensors is unfolding, also known as flattening or matricization. This operation transforms a tensor into a matrix by taking the fibers along a certain mode of the tensor and arranging them as columns of a matrix. The unfolding of a tensor $\mathcal{X}$ along mode N is denoted $\mathcal{X}_{(N)}$. In this process, an element $\mathcal{X}_{i_1, i_2, \dots, i_{N-1}, i_N, i_{N+1}, \dots, i_n}$ maps to a matrix element $\mathcal{X}_{(N)}(i_N, j)$, where

$$j = 1 + \sum_{k=1, k \neq N}^n \left((i_k - 1) \prod_{p=1, p \neq N}^{k-1} m_p \right). \quad (2.4)$$

Tensor unfolding allows us to use well-established tools from matrix analysis on tensors. For instance, N -mode product can be mapped to a matrix-matrix

multiplication using unfolding:

$$\mathbf{Y} = \mathbf{X} \times_N \mathbf{U} \Leftrightarrow \mathbf{Y}_{(N)} = \mathbf{U} \mathbf{X}_{(N)}, \quad (2.5)$$

where $\mathbf{Y}$ can be obtained by folding $\mathbf{Y}_{(N)}$ (i.e. by performing the inverse of unfolding).

The n D tensor $\mathbf{X}$ is *rank one* if it can be written as $\mathbf{X} = \mathbf{a}^{(1)} \circ \mathbf{a}^{(2)} \circ \dots \circ \mathbf{a}^{(n)}$, where $\mathbf{a}^{(i)} \in \mathbb{R}^{m_i}$ and the symbol $\circ$ denotes the outer product of vectors. Moreover, the rank of $\mathbf{X}$ is the smallest positive integer R such that

$$\mathbf{X} = \sum_{r=1}^R \mathbf{a}^{(r,1)} \circ \mathbf{a}^{(r,2)} \circ \dots \circ \mathbf{a}^{(r,n)}. \quad (2.6)$$

Calculating the rank of a tensor is NP-hard. This is in contrast with the rank of a matrix, which is uniquely defined and can be obtained by e.g. Singular Value Decomposition (SVD).

The tensor rank decomposition in (2.6) is known as the CANDECOMP/PARAFAC decomposition, or CP in short. While CP decomposition requires weaker conditions for uniqueness [114] (in contrast to matrix rank decompositions), it is typically intractable to compute. This is because R is unknown and an algorithm that solves (2.6) for a fixed R remains an active research problem [115, 116, 117, 118, 119, 120]. Another tensor decomposition method that has been widely used in many scientific and engineering problems is the Higher Order SVD (HOSVD) [121], also known as the Tucker decomposition. The HOSVD of the tensor $\mathbf{X}$ is as follows

$$\mathbf{X} = \mathbf{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \dots \times_n \mathbf{U}^{(n)}, \quad (2.7)$$

where $\mathbf{G}$ is called a *core tensor* and the matrices $\{\mathbf{U}^{(i)}\}_{i=1}^n$ are orthogonal (often orthonormal). It is possible to obtain a truncated HOSVD by only computing a fraction of columns of $\{\mathbf{U}^{(i)}\}_{i=1}^n$. In this way, $\mathbf{G}$ becomes a compressed version of $\mathbf{X}$. Alternatively, one can set the small values of $\mathbf{G}$ to zero to achieve compression of $\mathbf{X}$. It should be noted that unlike truncated SVD of matrices, truncated HOSVD is not necessarily optimal in an ℓ_2 sense. However, computing HOSVD is straightforward, see Algorithm 1.

While the result of HOSVD is not necessarily optimal, it is typically used as a starting point for iterative algorithms such as Alternating Least Squares (ALS) [115].

In some applications, it is more convenient to use the matrix representation of the N -mode product. This enables the utilization of existing tools in matrix analysis.

Input: A tensor $\mathcal{X} \in \mathbb{R}^{m_1 \times m_2 \times \dots \times m_n}$ and the desired rank of factor matrices, $(r_1, \dots, r_n)$, where $r_1 \leq m_1, \dots, r_n \leq m_n$.

Result: Core tensor $\mathcal{G} \in \mathbb{R}^{m_1 \times m_2 \times \dots \times m_n}$ and factor matrices $\{\mathbf{U}^{(i)} \in \mathbb{R}^{m_i, r_i}\}_{i=1}^n$

```

1 for  $i = 1$  to  $n$  do
2    $\mathcal{X}_{(i)} \leftarrow$  unfold  $\mathcal{X}$  along mode  $i$  ;
3   Compute SVD:  $\mathcal{X}_{(i)} = \mathbf{U}^{(i)} \mathbf{S} \mathbf{V}$ ;
4    $\mathbf{U}^{(i)} \leftarrow$  leading left  $r_i$  columns from  $\mathbf{U}^{(i)}$ ;
5 end
6  $\mathcal{G} \leftarrow \mathcal{X} \times_1 (\mathbf{U}^{(1)})^T \times_2 (\mathbf{U}^{(2)})^T \times_3 \dots \times_n (\mathbf{U}^{(n)})^T$ ;

```

Algorithm 1: Higher Order Singular Value Decomposition (HOSVD)

An important mathematical operation in this regard is the Kronecker product, defined for two matrices $\mathbf{A} \in \mathbb{R}^{m_1, m_2}$ and $\mathbf{B} \in \mathbb{R}^{p_1, p_2}$ as

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} \mathbf{A}_{1,1} \mathbf{B} & \dots & \mathbf{A}_{1,m_2} \mathbf{B} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{m_1,1} \mathbf{B} & \dots & \mathbf{A}_{m_1,m_2} \mathbf{B} \end{bmatrix}. \quad (2.8)$$

Using the Kronecker product, we can rewrite (2.7) in the matrix form as

$$\mathcal{X}_{(i)} = \mathbf{U}^{(i)} \mathcal{G}_{(i)} \left(\mathbf{U}^{(n)} \otimes \dots \otimes \mathbf{U}^{(i+1)} \otimes \mathbf{U}^{(i-1)} \otimes \dots \otimes \mathbf{U}^{(1)} \right)^T, \quad (2.9)$$

or in a vectorized form as

$$\text{vec}(\mathcal{X}) = \left(\mathbf{U}^{(n)} \otimes \dots \otimes \mathbf{U}^{(1)} \right) \text{vec}(\mathcal{G}). \quad (2.10)$$

2.3 Sparse Signal Estimation

The goal of sparse signal estimation is to calculate the sparse representation of a signal given a dictionary. The algorithms performing this task are sometimes called *sparse coding* or *sparse signal recovery* algorithms. Let us first formulate the problem for a 1D signal $\mathbf{y} \in \mathbb{R}^m$. The formulation will later be expanded to higher dimensionalities. Assume that $\mathbf{y}$ is sparse in an overcomplete dictionary $\mathbf{D} \in \mathbb{R}^{m \times k}$; i.e. in the absence of noise we have $\mathbf{y} = \mathbf{D}\mathbf{s}$, where $\|\mathbf{s}\|_0 \leq \tau$ and τ is the sparsity. If the signal is noisy, denoted $\hat{\mathbf{y}}$, then the representation is not exact, i.e. $\hat{\mathbf{y}} \approx \mathbf{D}\mathbf{s}$. Alternatively, we may be able to find an exact representation $\hat{\mathbf{y}} = \mathbf{D}\hat{\mathbf{s}}$, however

at the cost of reducing sparsity, i.e. $\|\hat{\mathbf{s}}\|_0 \geq \|\mathbf{s}\|_0$. The sparse signal estimation is formulated as follows

$$\min_{\mathbf{s}} \|\mathbf{s}\|_0 \quad \text{s.t.} \quad \|\mathbf{y} - \mathbf{D}\mathbf{s}\|_2^2 \leq \epsilon, \quad (2.11)$$

where an estimate of noise is required to set the value of ϵ . If we have an estimate of sparsity, we may solve

$$\min_{\mathbf{s}} \|\mathbf{y} - \mathbf{D}\mathbf{s}\|_2^2 \quad \text{s.t.} \quad \|\mathbf{s}\|_0 \leq \tau. \quad (2.12)$$

When neither τ nor ϵ can be estimated, the unconstrained problem (i.e. the Lagrangian form) is solved

$$\min_{\mathbf{s}} \lambda \|\mathbf{s}\|_0 + \frac{1}{2} \|\mathbf{y} - \mathbf{D}\mathbf{s}\|_2^2. \quad (2.13)$$

Equation (2.13) is known as the Basis Pursuit DeNoising (BPDN) [122]. Solving the above problems is NP-hard due to non-convexity of the ℓ_0 norm. The main difficulty arises from the fact that we do not know the location of nonzero values in $\mathbf{s}$. Let us define the function $\text{supp}(\mathbf{s})$, which outputs the location of nonzero values as an index set, known as the *support* of a sparse signal. In fact, if the support of $\mathbf{s}$ is known, then (2.11) reduces to a least squares problem with the closed-form solution

$$\begin{cases} \mathbf{s}_\Lambda = \mathbf{D}_\Lambda^\dagger \mathbf{y}, \\ \mathbf{s}_{\{1, \dots, k\} \setminus \Lambda} = 0, \end{cases} \quad (2.14)$$

where the operator $\setminus$ denotes set difference. However, the support of $\mathbf{s}$ is in general unknown and as of writing this thesis, there is no algorithm other than an exhaustive search over all possible support sets; i.e. one has to solve $\binom{k}{\tau}$ least squares problems, assuming that we know the sparsity of the solution. As an example, finding the support for an image of size 32×32 that is sparse, with $\tau = 16$, under a dictionary $\mathbf{D} \in \mathbb{R}^{1024 \times 2048}$ will take approximately $1.37\text{e}+22$ years to complete if each least squares problem takes $1.0\text{e}-10$ of a second!

The non-convexity problem of the ℓ_0 norm can be addressed by replacing it with an ℓ_1 norm. In this way, the optimization problem becomes convex and one can rewrite it such that classical linear programming algorithms can be used [76]. In Fig. 2.1 a geometrical interpretation of the solution of (2.11) for different types of norm when $k = 3$ is shown. The solution, i.e. $\mathbf{s}$, is at the intersection of the feasible set defined by $\mathbf{y} = \mathbf{D}\mathbf{s}$ and a ball defined by the norm of $\mathbf{s}$. The feasible set forms a hyperplane of dimension $\mathbb{R}^{k-m}$ embedded in $\mathbb{R}^k$ [66]. Comparing different norms in Fig. 2.1, we see that the ℓ_2 norm does not produce a sparse solution. This is because the intersection point is not on a coordinate axis. In contrast, for ℓ_0 and ℓ_1 norms the solution lies on a coordinate axis, hence the other two components of $\mathbf{s}$ are zero. More details about the geometric interpretation of the sparse recovery problem can be found in [63, 123, 124].

Figure 2.1: Intersection of the feasible set (the yellow plane) with the (a) ℓ_0 ball, (b) ℓ_1 ball, and (c) ℓ_2 ball. The intersection point is shown in red.

There exists a large body of research on deriving algorithms for sparse signal estimation. Indeed finding more efficient algorithms remains an active research topic. Sparse recovery algorithms can be divided into two main categories depending on whether the ℓ_0 or the ℓ_1 variant is being considered.

The first category is *Greedy* methods, which estimate the support of the signal using iterative procedures instead of solving an optimization problem. Matching Pursuit (MP) [125] and Orthogonal Matching Pursuit (OMP) [126] are probably the earliest examples of greedy methods for solving (2.11). These algorithms estimate one new support index at each iteration. Moreover, due to the heuristic nature of MP and OMP, deriving theoretical guarantees is challenging. One of the contributions of this thesis is a theoretical analysis on the support recovery performance of OMP, see Section 5.7 and Paper F. Many algorithms have been proposed to improve the performance of MP and OMP. For instance, Regularized OMP (ROMP), Compressive Sampling MP (CoSaMP) [127], and Stagewise OMP [128] recover multiple support indices at each iteration and offer stronger theoretical guarantees. Other examples of greedy algorithms are [129, 130, 131, 132, 133, 134, 135, 136, 137].

The second category contains *convex relaxation* methods, also known as optimization based or ℓ_1 algorithms. These methods are based on different variations of linear programming for solving an ℓ_1 problem. Moreover, convex relaxation algorithms offer guaranteed convergence due to the convexity of the problem. However, they are substantially slower than greedy algorithms. For instance, OMP and ROMP have a running time complexity of $\mathcal{O}(\tau mk)$, while linear programming using the interior point method has a complexity of $\mathcal{O}(m^2 k^{1.5})$, see [127]. Additionally, worst case time complexity of ℓ_1 methods is exponential, while greedy methods are typically linear. A few examples of ℓ_1 algorithms include GPSR [138], SPGL1 [139], SpaRSA [140], LARS [141], and [142, 143, 144, 145, 146, 147]; see also [148] for a classification of different ℓ_1 solvers.

Apart from the two categories described above, there exists a “hybrid” family of methods. For instance, Smoothed- ℓ_0 (SL0) [82] approximates the ℓ_0 pseudo-norm with a continuous function controlled by a parameter α that converges to the ℓ_0 pseudo-norm when $\alpha \rightarrow 0$. While the method directly solves the ℓ_0 problem, it is based on gradient descent instead of a greedy algorithm. Other examples of hybrid methods include [149, 150, 151, 152, 153, 154, 155].

All of the algorithms mentioned above solve a sparse recovery problem for a vector, i.e. a 1D signal. While sparse recovery of 1D signals have received a lot of attention during the last two decades, recovery of high dimensional signals, i.e. tensors, have been relatively untouched. Cai and Cichocki propose the Kronecker-OMP algorithm [102, 156], a modification of the OMP method for tensors. Moreover, they also propose N-BOMP, an algorithm designed for block-sparse [157] tensors. In [158], the authors solve a nonlinear problem with the assumption of low rank or sparse structures along each mode. Moreover, a recovery method for low-rank tensors synthesized as sums of outer products of sparse loading vectors was proposed in [159]; however, the algorithm relies on solving the NP-hard problem of estimating the tensor rank. More recently, Friedland et al. proposed Generalized Tensor Compressed Sensing (GTCS) [160]. The method is shown to be vastly superior in computational complexity compared to vectorized sparse recovery but suffers from poor reconstruction quality in real world applications [161].

2.4 Dictionary Learning

The goal of dictionary learning is to construct a set of basis functions using machine learning such that any input signal can be modeled as a linear combination of a small subset of basis functions with the least amount of error. Just like any machine learning method, one needs a large training set that is representative of the signal family being considered (e.g. a large number of images). If the training set does not capture different variations of the data, sparse representation of the unobserved test signals leads to low sparsity, or large error, or both.

Since sparse representation is the goal of dictionary learning, one needs an algorithm for sparse recovery as described in Section 2.3. On the other hand, the dictionary is a parameter in the sparse recovery problem, see (2.11). In such “chicken-and-egg” problems, it is a common practice to iteratively solve an optimization problem for one of the parameters while fixing the other until the convergence of both parameters. For instance, one can start by a random initialization of the dictionary, solve for the sparse coefficients, update the dictionary, and so on.

Perhaps the most commonly used dictionary learning method is K-SVD [162]. Moreover, this algorithm is typically used as a baseline for evaluating the effective-

ness of dictionary learning methods. Since a large family of algorithms follow the same steps as K-SVD, albeit with different approaches for solving the problem, we start by describing this method. The K-SVD algorithm solves the following optimization problem

$$\min_{\mathbf{S}, \mathbf{D}} \|\mathbf{X} - \mathbf{DS}\|_2^2 \quad \text{s.t.} \quad \|\mathbf{S}_i\|_0 \leq \tau, \forall i \in \{1, \dots, N\}, \quad (2.15)$$

where $\mathbf{X} \in \mathbb{R}^{m \times N}$ is a collection of N training signals, and τ is a user-defined sparsity parameter; moreover, $\mathbf{D} \in \mathbb{R}^{m \times k}$ and $\mathbf{S} \in \mathbb{R}^{k \times N}$ are the dictionary and sparse coefficients that we would like to estimate, respectively. A dictionary with $k > m$ is called *overcomplete*. The larger the number of columns in $\mathbf{D}$, the higher the sparsity of the representation. A dictionary that can model a larger group of signals with high sparsity and low error is called *representative*. While increasing k leads to a more representative dictionary, it comes at the cost of significantly higher computational cost per iteration and a slower convergence. Moreover, increasing the overcompleteness requires a larger training set. It is common to use $k = \delta m$, where δ is a scalar defining overcompleteness. For instance, if $\delta = 2$, we call the dictionary two times overcomplete.

K-SVD iterates over two steps until convergence: 1. *sparse coding* and 2. *dictionary update*. The sparse coding is done using OMP for each signal in the training set by solving

$$\min_{\mathbf{S}_i} \|\mathbf{X}_i - \mathbf{DS}_i\|_2^2 \quad \text{s.t.} \quad \|\mathbf{S}_i\|_0 \leq \tau. \quad (2.16)$$

Alternatively, equation (2.16) can be solved using Batch-OMP [163] because $\mathbf{D}$ is fixed for all $\mathbf{X}_i$.

The dictionary update is done one atom at a time, i.e. by fixing all the atoms except one. In order to isolate the i th atom for updating, we can rewrite the objective function of (2.15) as

$$\|\mathbf{X} - \mathbf{DS}\|_2^2 = \|\mathbf{E}^{(i)} - \mathbf{D}_i \mathbf{S}_{i,\cdot}\|_2^2, \quad \text{where} \quad \mathbf{E}^{(i)} = \mathbf{X} - \sum_{j=1, j \neq i}^m \mathbf{D}_j \mathbf{S}_{j,\cdot}, \quad (2.17)$$

The vector $\mathbf{S}_{i,\cdot} \in \mathbb{R}^N$ is the i th row of $\mathbf{S}$ and describes which signals in the training set use the i th atom. The term $\mathbf{E}^{(i)}$ is a residual term that we would like to estimate using atom i . Indeed the solution of (2.17) is obtained by a rank one approximation of $\mathbf{E}^{(i)}$ using SVD. However, in this way $\mathbf{S}_{i,\cdot}$ will not fulfill the sparsity requirement. To account for the fact that we seek a sparse coefficient matrix $\mathbf{S}$, we limit (2.17) to the elements of $\mathbf{E}^{(i)}$ and $\mathbf{S}_{i,\cdot}$ that are in the support set of $\mathbf{S}_{i,\cdot}$; i.e. we only update the nonzero values of $\mathbf{S}_{i,\cdot}$, which means only the signals that use the atom i contribute to updating it. This can be simply achieved by multiplying (2.17) with

a sampling matrix that samples elements according to $\text{supp}(\mathbf{S}_{i,\cdot})$. Hence, equation (2.17) becomes

$$\|\mathbf{E}^{(i)} - \mathbf{D}_i \mathbf{S}_{i,\cdot}\|_2^2 \propto \|\mathbf{E}^{(i)} \boldsymbol{\Omega}^{(i)} - \mathbf{D}_i \mathbf{S}_{i,\cdot} \boldsymbol{\Omega}^{(i)}\|_2^2 = \|\boldsymbol{\Upsilon} - \mathbf{D}_i \boldsymbol{\rho}^{(i)}\|_2^2, \quad (2.18)$$

where $\boldsymbol{\Upsilon} = \mathbf{E}^{(i)} \boldsymbol{\Omega}^{(i)}$ and $\boldsymbol{\rho}^{(i)} = \mathbf{S}_{i,\cdot} \boldsymbol{\Omega}^{(i)}$. We can now compute a rank one SVD as follows $\boldsymbol{\Upsilon} = \sigma \mathbf{u} \mathbf{v}^T$, and the minimizer of (2.18) is obtained by setting $\mathbf{D}_i = \mathbf{u}$ and $\boldsymbol{\rho}^{(i)} = \sigma \mathbf{v}$.

Since the advent of K-SVD, signal and image processing communities have been actively reporting new and improved dictionary learning methods. Online dictionary learning [164] uses stochastic optimization techniques to significantly accelerate the performance of K-SVD. An ℓ_1 variant of K-SVD was proposed in [165] by adopting a probabilistic approach. A different approach for solving (2.15) was introduced in [166] that uses majorization optimization approach, with guaranteed convergence. Maill et al. propose INK-SVD by introducing an intermediate step in K-SVD that decorrelates dictionary atoms. Low coherence leads to an improved compressed sensing performance, see Chapter 5. The proposed method was improved in [167]. A method for training dictionaries that are invariant under rotation and flipping was proposed in [168]. However, the method learns the invariance for a pre-defined set of transforms. A different criteria for dictionary learning was proposed in [169], where instead of minimizing the sparsity, a low entropy representation is sought for compression. An extension of K-SVD to handle additive sparse corruptions of training signals was proposed in [170]. A modification of K-SVD was discussed in [171], where similar sparsity for individual signals is promoted rather than a global sparsity as in (2.15). Another enhancement of K-SVD comes in the form of adaptively determining the required number of atoms during training [172, 173, 174, 175].

There exists a family of “task-driven” dictionary learning methods that tune the dictionary for a specific task, e.g. classification and clustering [176, 177, 178, 179, 180, 181, 182], image denoising [183, 184, 185, 186], medical imaging [187, 188, 189], plenoptic imaging [190], or even reducing the block artifacts of JPEG images [191]. In [192], the authors propose a variant of K-SVD where only the dictionary update is performed. Moreover, unlike (2.18) where only the coefficients in the support are updated, the new method updates the entire vector $\mathbf{S}_{i,\cdot}$, hence admitting the change of support in each iteration. A divide-and-conquer method was used in [193] to train dictionaries for correlated training examples, followed by a method for combining the atoms. In [194], an algorithm for directly solving (2.15) without an alternating approach is presented; however, the performance of the proposed method is not competitive with K-SVD [195]. Liang et al. [196] propose a distributed algorithm to learn a dictionary over multiple nodes that can be used in a sensor network

environment. More recently, convolutional neural networks have been utilized for dictionary learning and sparse representations [197].

Multidimensional dictionary learning has only recently attracted researchers for proposing methods that can take as input matrices and tensors. In classical dictionary learning, i.e. one that is formulated by (2.15), multidimensional signals in the training set are vectorized. However, there are two major shortcomings with this approach. First, the trained atoms do not capture structures present along each dimension independently. For instance, in a light field video each dimension possesses simple structures that can be utilized for training atoms. However, when these structures or similarities are combined in a single object (i.e. a vector), the model becomes too complicated for training. Second, vectorizing a multidimensional signal often leads to a very large vector, hence dramatically increasing the training time. In many cases, this issue makes the training for large datasets intractable.

Let $\{\mathcal{X}^{(i)}\}_{i=1}^N$ be a collection of N tensors with a dimensionality of $m_1 \times \cdots \times m_n$. The multidimensional dictionary learning problem can be formulated as

$$\begin{aligned} \min_{\hat{\mathcal{S}}, \{\mathbf{U}^{(j)}\}_{j=1}^n} & \left\| \hat{\mathcal{X}} - \hat{\mathcal{S}} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \cdots \times_n \mathbf{U}^{(n)} \right\|_F^2 \\ \text{s.t. } & \left\| \hat{\mathcal{S}}_{:, \dots, :, i} \right\|_0 \leq \tau, \forall i \in \{1, \dots, N\}, \end{aligned} \quad (2.19)$$

where $\hat{\mathcal{X}} \in \mathbb{R}^{m_1 \times \cdots \times m_n \times N}$ is the collection of N tensors for training arranged along the last dimension. Similarly, $\hat{\mathcal{S}} \in \mathbb{R}^{k_1 \times \cdots \times k_n \times N}$ contains N sparse coefficient tensors. Moreover, the dictionary is a collection of n matrices $\{\mathbf{U}^{(j)} \in \mathbb{R}^{m_j \times k_j}\}_{j=1}^n$. As before, if $m_j < k_j$, then the dictionary is overcomplete along the j th dimension. To the best of author's knowledge, there exists only a few methods for solving (2.19). K-HOSVD and T-MOD were proposed in [198], which are extensions of K-SVD and MOD [199] for tensors. However, the training algorithm is only evaluated on a synthetic dataset. Moreover, the method requires $k_1 k_2 \dots k_n$ updates of dictionary atoms. In other words, each atom of $\{\mathbf{U}^{(j)}\}_{j=1}^n$ is updated multiple times. Separable Dictionary Learning (SeDiL), proposed in [200], trains a dictionary of the form $\mathbf{D} = \mathbf{U}^{(2)} \otimes \mathbf{U}^{(1)}$ while enforcing low coherence on the dictionary. The authors solve the problem using algorithms for optimization on manifolds [201]. However, it is unclear how the method can be extended to higher dimensions. Recently, Ding et al. [104] propose a method for solving (2.19) that requires $k_1 + k_2 + \cdots + k_n$ atom updates, although with an additional least squares problem for each update. The authors also present a method for joint optimization of the multidimensional dictionary and a sensing matrix for compressed sensing.

Figure 2.2: Light field parametrization.

2.5 Visual Data: The Curse/Blessing of Dimensionality

Perhaps the most simple type of visual data is an image, defined by the function $i(x, y, \lambda)$, where (x, y) defines the location of a pixel and λ defines the wavelength. Images with three wavelength bands (red, green, and blue) are omnipresent in our daily life. It is also possible to capture and store images with more than three wavelength bands, which are called *multispectral* or *hyperspectral*. Multispectral images have a wide range of applications in agriculture, environmental monitoring, pharmaceuticals, and industrial quality control. If we add a temporal dimension to an image, we obtain a video, defined by $v(x, y, \lambda, t)$. A simple camera with a lens captures an image by integrating incoming radiance from multiple directions on a pixel during the short period of time the shutter is open. The depth of field is a byproduct of this integration. Multispectral imaging requires more advanced setups, see e.g. [103, 202, 203, 204].

While an image only contains spatial (projected) luminance values from a scene, a

light field contains both spatial and angular information. If we define a light field on a plane, just like an image, then each pixel has multiple values corresponding to outgoing radiance along directions defined by a hemisphere centered at the pixel. This can be described by the function $l(x, y, \phi, \theta, \lambda)$, where (ϕ, θ) defines a direction vector in spherical coordinates (see Fig. 2.2a). A different parametrization is based on defining two planes for camera and image [20], defined by the function (r, t, u, v, λ) , where (r, t) and (u, v) parameterize the camera and image planes, respectively (see Fig. 2.2b). Furthermore, a light field can be described over a 3D geometrical object [205], which is known as *surface light field*. At each point on an object, we place a hemisphere aligned with the normal at the point. The outgoing radiance values along multiple directions in this hemisphere are stored, which can be described by the function $l(x, y, z, \phi, \theta, \lambda)$. There exists other types of light field parametrizations [206, 207, 208]. Indeed one can append a temporal dimension to any of the above mentioned parametrizations. The result is called a *light field video*, defined by e.g. $l(x, y, \phi, \theta, \lambda, t)$. Additionally, multispectral light field cameras have emerged recently [209, 210]. In Chapter 1 we mentioned datasets that describe reflectance properties, e.g. BRDF, SVBRDF, and BTF. The common denominator for large-scale multidimensional datasets is the extremely tedious tasks of acquisition, storage, and processing. Regarding storage, a single-precision floating-point light field with 4K UHD spatial resolution and an angular resolution of 32×32 consumes 97.2 gigabytes of storage. An hour long video of this light field requires about 2.15 petabytes of storage at 30 frames per second. Indeed the acquisition of such datasets also pose many challenges due to the limitations of sensors and the required bandwidth. Moreover, direct processing on the entire dataset is typically not possible. These challenges are associated with the *curse of dimensionality*.

To tackle the problem of multidimensional visual data processing, a common approach is to divide the dataset into small elements. We call such an element from a visual dataset a *data point*, or a *signal* depending on the context. Let us provide a few examples and setup the notations that will be used in the remainder of this thesis. Let the function $l(\mathbf{r}_i, \mathbf{t}_j, \mathbf{u}_\alpha, \mathbf{v}_\beta, \lambda_\gamma)$ describe a discrete two plane light field. We divide the light field into a set of data points $\{\mathcal{X}^{(i)}\}_{i=1}^N$, where $\mathcal{X}^{(i)} \in \mathbb{R}^{m_1 \times m_2 \times m_3 \times m_4 \times m_5}$. In other words, we construct small patches of size $m_1 \times m_2$ over the camera plane $(\mathbf{r}_i, \mathbf{t}_j)$, and patches of size $m_3 \times m_4$ over the image plane $\mathbf{u}_\alpha, \mathbf{v}_\beta$, as well as 1D patches of size m_5 along the spectral domain. One can also obtain 1D, 2D, 3D, and 4D data points from a 5D dataset by unfolding certain dimensions. For instance, a 2D data point from a light field can be defined as $\mathbf{X}^{(i)} \in \mathbb{R}^{m_1 m_2 \times m_3 m_4 m_5}$, i.e. the rows contain an unfolded camera plane patch and the columns contain an unfolded color patch from the image plane. We assume that the dimensionality of data points are fixed over the entire dataset.

In practice, for a seamless angular representation of the light field, we include all the angular and spectral information in each data point, while dividing the spatial domain into small patches. A light field video can be subdivided into small data points in a similar fashion, with the addition that we divide the time domain into small segments, each containing a few consecutive frames. Hence we obtain 6D data points $\mathcal{X}^{(i)} \in \mathbb{R}^{m_1 \times m_2 \times m_3 \times m_4 \times m_5 \times m_6}$, where m_6 is the number of consecutive frames used in each data point.

Dividing a multidimensional dataset into small data points has several benefits. First, it enables parallel processing of the dataset, as well as bypassing the memory requirements of processing the entire dataset. More importantly, for reasons that will be described in Section 3.2, visual datasets are known to exhibit coherence or similarities among small data points. These similarities occur both in the original domain of the data point, and in the transformed domain, e.g. when we use sparse representation of the data points. We call this family of similarities *inter data point coherence*. Moreover, there exist similarities inside each data point, which we call *intra data point coherence*. These two properties are fundamental elements that describe the superior performance of the methods proposed in this thesis for efficient compression and compressed sensing compared to previous work. We call this the *blessing of dimensionality*, because as the dimensionality of the dataset grows, specifically for visual data, we observe more and more inter and intra coherence among data points.

Learning-based Sparse Representation of Visual Data

In Section 2.5 we described the notion of the *blessing of dimensionality*. As the dimensionality of the data grows, the amount of self-similarities (also known as coherence) in the data grows. There are numerous approaches for exploiting the blessing of dimensionality [211]. For instance, we may be able to find a low-dimensional representation that accommodates the data with minimal error. Linear variants of such methods are known as low-rank approximations [212] and non-linear algorithms are referred to as manifold learning [213, 214, 215]. A prominent tool for exploiting the blessing of dimensionality is sparse representations, where the data is modeled using a dictionary such that only a few atoms are needed for an accurate representation, as described in Section 1.1. We also discussed in Section 1.1 that one of the applications of sparse representations is compression. A high-level flowchart for compression using a training-based dictionary is illustrated in Fig. 3.1 below.

This thesis makes contributions on the sparse representation of visual data for compression using a new dictionary learning algorithm. In particular, this chapter will introduce methods for training an ensemble of multidimensional dictionaries that enable highly sparse representations with a small representation error. The outline of this chapter, as well as the contributions of this thesis on the topic of sparse representations, will be presented in Section 3.1. Applications of the algorithms presented in this chapter for the compression of multidimensional visual data will be presented in Chapter 4.

Figure 3.1: A flowchart for compression using a training-based sparsifying dictionary.

3.1 Outline and Contributions

Section 3.2 presents a method for learning a Multidimensional Dictionary Ensemble (MDE). This model enables highly sparse representations of visual data, while the size of the ensemble is orders of magnitude smaller than competing dictionary learning methods such as K-SVD. Since each dictionary in the ensemble is orthonormal, given a dataset for compression, a fast greedy algorithm is used to determine the best dictionary in the ensemble. This method was first introduced in [105, 216], and in Paper B we extend the learning algorithm to accommodate n D datasets. This extension is important since existing visual data, as described in Chapter 1, vary in dimensionality, e.g. from 1D to 7D. Moreover, it is expected that the dimensionality and complexity of visual data to increase in the near future as new modalities for capturing natural phenomena emerge. Additionally, Paper B presents a novel pre-clustering approach that improves the quality of representation, while decreasing the training time by an order of magnitude (see Section 3.3). We call the ensemble obtained from this method an Aggregate MDE, or AMDE in short. The improvement in training time and representation power enables AMDE to be used for a variety of applications involving high dimensional visual data that have a prohibitive memory footprint.

3.2 Multidimensional Dictionary Ensemble

This section describes a training based method for computing a Multidimensional Dictionary Ensemble (MDE). Our motivation for choosing this model for training, as well as its benefits for sparse representation of visual data will be presented in Section 3.2.1. Moreover, we state the requirements of a suitable dictionary for compression, and compressed sensing, with respect to visual data. In Section 3.2.2, we formulate an optimization problem for training an MDE that satisfies the requirements described in Section 3.2.1. Additionally, a brief explanation of our approach for solving the optimization problem is presented. Finally, in Section

Table 3.1: Table of variables and parameters used for training an AMDE.

symbol	description	dimension
n	# of dimensions for a data point	scalar
$m_1, \dots, m_n$	n D data point dimensionality	set
N_l	# of training data points	scalar
N_t	# of testing data points	scalar
K	# of dictionaries in the ensemble	scalar
τ_l	training sparsity	scalar
τ_t	testing sparsity	scalar
$\mathcal{L}^{(i)}$	i th training data point	$\mathbb{R}^{m_1 \times \dots \times m_n}$
$\mathcal{T}^{(i)}$	i th testing data point	$\mathbb{R}^{m_1 \times \dots \times m_n}$
$\mathbf{U}^{(j,k)}$	j th dictionary element of k th dictionary	$\mathbb{R}^{m_j \times m_j}$
$\mathcal{S}^{(i)}$	i th sparse coefficients tensor	$\mathbb{R}^{m_1 \times \dots \times m_n}$
$\mathbf{M}$	training membership matrix	$\mathbb{R}^{N_l \times K}$
$\mathbf{m}$	membership vector of MDE	$\mathbb{R}^{N_l \times K}$

3.2.3, an algorithm for obtaining sparse coefficients of data points we would like to compress will be described. For convenience, Table 3.1 includes the parameters and variables used throughout this section for training an MDE and utilizing it for compression.

3.2.1 Motivation

Indeed there exists numerous possibilities for dictionary training depending on the input dataset. Hence, before delving into the details of MDE, we present the motivation behind our proposed model for sparse representation of visual data. Let us consider a training set $\{\mathbf{L}^{(i)}\}_{i=1}^{N_l}$, where $\mathbf{L}^{(i)} \in \mathbb{R}^{m_1 \times m_2}$ and N_l is the number of data points in the training set. Such a training set can be obtained from images, or even higher dimensional datasets if we unfold a tensor into a matrix (see Section 2.5). Applying Singular Value Decomposition (SVD) on each data point yields $\mathbf{L}^{(i)} = \mathbf{U}^{(i)} \mathbf{S}^{(i)} (\mathbf{V}^{(i)})^T$, where $\mathbf{U}^{(i)} \in \mathbb{R}^{m_1 \times m_1}$ and $\mathbf{V}^{(i)} \in \mathbb{R}^{m_2 \times m_2}$ are an orthonormal basis pair containing left and right singular vectors, respectively. In addition, the diagonal matrix $\mathbf{S}^{(i)} \in \mathbb{R}^{m_1 \times m_2}$ contains the singular values in a decreasing order. The number of nonzero elements in $\mathbf{S}^{(i)}$ defines the rank of $\mathbf{L}^{(i)}$. From a different perspective, SVD defines orthogonal dictionaries $\mathbf{U}^{(i)}$ and $\mathbf{V}^{(i)}$, along with a sparse coefficient matrix $\mathbf{S}^{(i)}$. The number of nonzero elements of $\mathbf{S}^{(i)}$ defines the sparsity of $\mathbf{L}^{(i)}$. This decomposition is exact and always exists.

For n D data points, $\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}$, where $\mathbf{L}^{(i)} \in \mathbb{R}^{m_1 \times \dots \times m_n}$, a rank-revealing decomposition that also produces orthogonal dictionaries does not exist. A close candidate for this task is Higher Order SVD (HOSVD) [121] which computes the decomposition

$$\mathcal{L}^{(i)} = \mathcal{S}^{(i)} \times_1 \mathbf{U}^{(1,i)} \dots \times_n \mathbf{U}^{(n,i)} = \mathcal{S}^{(i)} \bigtimes_{j=1}^n \mathbf{U}^{(j,i)}, \quad (3.1)$$

where $\mathbf{U}^{(j,i)} \in \mathbb{R}^{m_j \times m_j}$, $j = 1, \dots, n$, are orthogonal dictionaries for $\mathcal{L}^{(i)}$; the coefficient tensor $\mathcal{S}^{(i)} \in \mathbb{R}^{m_1 \times \dots \times m_n}$ is all-orthogonal and describes the linear relation between the dictionaries and the data point. Moreover, the symbol $\times_j$ defines tensor-matrix multiplication along the j th mode; see Section 2.2 for more details. We can view the collection of orthonormal matrices $\{\mathbf{U}^{(j,i)}\}_{j=1}^n$ as basis matrices along different modes and the elements of the core tensor ($\mathcal{S}^{(i)}$) as coefficients describing the linear combination of bases to form $\mathcal{L}^{(i)}$.

In the case of the SVD, if a low-rank approximation is needed rather than an exact decomposition, one can nullify elements of $\mathcal{S}^{(i)}$ with the smallest absolute value. We denote the resulting matrix as $\hat{\mathcal{S}}^{(i)}$; hence, the approximated data point is $\hat{\mathbf{L}}^{(i)} = \mathbf{U}^{(i)} \hat{\mathcal{S}}^{(i)} (\mathbf{V}^{(i)})^T$. Here, the approximation error is defined as $\|\mathbf{L}^{(i)} - \hat{\mathbf{L}}^{(i)}\|_F$, which is equivalent to $\|\mathcal{S}^{(i)} - \hat{\mathcal{S}}^{(i)}\|_F$ due to the invariance of the Frobenius norm under orthonormal transformations. By Eckart-Young theorem [217], the low-rank approximation described above has the lowest error achievable with respect to any matrix norm. This approximation method can be extended to higher dimensions via HOSVD; i.e. by nullifying smallest elements in $\mathcal{S}^{(i)}$, we obtain an approximation of $\mathcal{L}^{(i)}$. However, unlike SVD, here the number of nonzero elements of $\mathcal{S}^{(i)}$ does not indicate the rank of $\mathcal{L}^{(i)}$. Tensor approximation using HOSVD has shown to produce acceptable results in practice for computer graphics and visualization applications [28, 34, 96, 97, 98, 218, 219].

So far we have considered the decomposition of one data point. For compression of a dataset comprising of a large number of data points, two families of approaches exist. When we apply a tensor decomposition such as HOSVD on every data point in the training set, we obtain an ensemble of N_l orthonormal multidimensional dictionaries $\{\mathbf{U}^{(1,i)}, \mathbf{U}^{(2,i)}, \dots, \mathbf{U}^{(n,i)}\}_{i=1}^{N_l}$. The obtained model is highly localized, i.e. each data point is independently represented in a unique subspace of $\mathbb{R}^{m_1 \times m_2 \times \dots \times m_n}$ by the coefficients $\mathcal{S}^{(i)}$. It is evident that such a model has a larger storage cost than the input data. Therefore, compression is achieved by nullifying small elements in the coefficient tensor $\mathcal{S}^{(i)}$, see e.g. [34, 97, 98]. Note that the dictionary columns that correspond to nonzero coefficients in $\mathcal{S}^{(i)}$ should also be stored for each data point, which reduces the compression ratio drastically. At the other end of the spectrum, one can define a single dictionary, using e.g. Principal Component Analysis (PCA) [220], or its higher dimensional variants [221, 222]. In this model, *all* the data

Table 3.2: Storage cost comparison of different dictionary types.

dictionary type	storage cost
two times overcomplete K-SVD	$\prod_{j=1}^n m_j \times 2 \prod_{j=1}^n m_j$
2D MDE with K dictionaries	$K \left(\prod_{j=1}^q m_j \times \prod_{j=1}^q m_j + \prod_{j=q+1}^n m_j \times \prod_{j=q+1}^n m_j \right)$
n D MDE with K dictionaries	$K (m_1 \times m_1 + m_2 \times m_2 + \cdots + m_n \times m_n)$

points are represented in *one* low-dimensional subspace. As a result, the size of the dictionary is very small. However, visual data exhibit global nonlinearities, along with locally linear structures [223, 224, 225, 226, 227, 228, 229], which renders a single multidimensional dictionary inadequate for accurate sparse representations. In conclusion, there is a trade-off between the quality and the storage cost of sparse signal representations with respect to the number of dictionaries. The larger the number of dictionaries, the higher the quality of representation, however at the cost of increasing the storage requirements.

We use a model that is the middle ground between the aforementioned techniques, hence providing a (user-defined) trade-off between the quality of reconstruction and the compression ratio. In particular, we train an ensemble of $K \ll N_l$ orthonormal multidimensional dictionaries. A data point is then represented by one dictionary as follows

$$\mathcal{L}^{(i)} = \mathcal{S}^{(i)} \times_1 \mathbf{U}^{(1,k)} \times_2 \cdots \times_n \mathbf{U}^{(n,k)} = \mathcal{S}^{(i)} \bigtimes_{j=1}^n \mathbf{U}^{(j,k)}, \quad (3.2)$$

where $\mathbf{U}^{(j,k)} \in \mathbb{R}^{m_j \times m_j}$ is the j th *dictionary element* of k th dictionary. A dictionary element is a matrix that operates along a specific dimension of a tensor. We denote the MDE by $\Psi = \left\{ \mathbf{U}^{(1,k)}, \dots, \mathbf{U}^{(n,k)} \right\}_{k=1}^K$, where K is a user-defined parameter. Since our goal is to obtain efficient sparse representations of visual data, there exists several additional requirements that should be fulfilled:

1. *Sparsity*: This is the key property that leads to efficient compression and compressed sensing of visual data. As discussed in Section 1.1, because visual datasets are typically compressible and not sparse, it is the effectiveness of the dictionary that defines the sparsity and the representation error. Moreover, sparsity is the key factor in exploiting the intra data point coherence.
2. *Clustering*: This is a key component in exploiting the inter data point coherence of a training set. Since each data point in our model is represented by one dictionary, and we have $K \ll N_l$, the training algorithm also performs a clustering of data points. Another benefit of clustering is that it promotes

the sparsity in the representation since it is unlikely that the entire training set is sparse in one dictionary.

3. *Representativeness*: In practice, we would like to perform the training once, and use the ensemble on a *test set*. The test set is a collection of data points that we would like to compress, or perform compressed sensing on. Since the test set is typically much larger than the training set, the trained ensemble should be efficient enough to enable sparse representation of the unobserved data points in the test set. This property is not possessed by tensor decomposition based methods such as [34, 96, 97, 98, 230].
4. *Arbitrary dimensionality*: Since we consider visual data, which are very diverse in terms of dimensionality, the dictionary ensemble should enable sparse representations regardless of the data dimensionality.
5. *Small dictionary size*: Since the MDE is trained from a set of examples, it needs to be stored for compression and compressed sensing. This is in contrast to analytical dictionaries where the dictionary is evaluated during encoding or decoding. Therefore, the size of the MDE is important for compression. A key component in reducing the size of the MDE is multidimensionality. Tensor decomposition and 1D dictionaries have very large memory footprints, which makes the transmission of the dictionary to the decoder very challenging [34, 162, 230]. See Table 3.2 for a storage cost comparison of different dictionary types (and see Section 4.3 for numerical comparisons).
6. *Fast local reconstruction*: For real time rendering of e.g. light field and light field videos, efficient local reconstruction of the compressed data is essential. This property is achieved by the orthogonality of dictionaries (examples of the local real time reconstruction of visual data will be given in Chapter 4). Moreover, since the ensemble is small, the reconstruction process can be fully implemented on the GPU.

In the following section, we formulate the problem of training an MDE that fulfills the requirements stated above.

3.2.2 Training

Given the requirements stated in Section 3.2.1, and inspired by [105, 216], we formulate the problem of training an MDE that enables the sparse representation

of visual data as follows

$$\min_{\mathbf{U}^{(j,k)}, \mathcal{S}^{(i,k)}, \mathbf{M}} \sum_{i=1}^{N_l} \sum_{k=1}^K \mathbf{M}_{i,k} \left\| \mathcal{L}^{(i)} - \mathcal{S}^{(i,k)} \times_{j=1}^n \mathbf{U}^{(j,k)} \right\|_F^2, \quad \text{subject to} \quad (3.3a)$$

$$\left(\mathbf{U}^{(j,k)} \right)^T \mathbf{U}^{(j,k)} = \mathbf{I}, \quad \forall k = 1, \dots, K, \quad \forall j = 1, \dots, n, \quad (3.3b)$$

$$\left\| \mathcal{S}^{(i,k)} \right\|_0 \leq \tau_l, \quad (3.3c)$$

$$\sum_{k=1}^K \mathbf{M}_{i,k} = 1, \quad \forall i = 1, \dots, N_l. \quad (3.3d)$$

The objective function, i.e. (3.3a), measures the representation error over all data points and dictionaries. Because each data point is represented by one dictionary, we need a clustering matrix to enforce this structure on the objective function. We call the clustering matrix $\mathbf{M} \in \mathbb{R}^{N_l \times K}$ a *membership matrix*, a binary matrix associating each data point $\mathcal{L}^{(i)}$ to its corresponding dictionary $\{\mathbf{U}^{(1,k)}, \dots, \mathbf{U}^{(n,k)}\}$ in the MDE. Each row in $\mathbf{M}$ has only one non-zero component to guarantee that a data point uses one dictionary; this is enforced by the constraint in (3.3d). The sparsity of the coefficient tensor $\mathcal{S}^{(i,k)}$ is enforced by the constraint in (3.3c), where τ_l is a user-defined parameter for sparsity. Moreover, equation (3.3b) puts an orthonormality constraint on the dictionaries. We denote the process of training an MDE by the following function

$$\{\mathbf{U}^{(1,k)}, \dots, \mathbf{U}^{(n,k)}\}_{k=1}^K = \mathbf{Train} \left(\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}, K, \tau_l \right). \quad (3.4)$$

Note that although the training stage computes $\mathbf{U}^{(j,k)}$, $\mathcal{S}^{(i,k)}$, and $\mathbf{M}$, we only need the MDE, i.e. $\Psi = \{\mathbf{U}^{(1,k)}, \dots, \mathbf{U}^{(n,k)}\}_{k=1}^K$. This is because the test set is different from the training set, and hence we need a new set of sparse coefficients and a membership matrix. Moreover, since we have three sets of unknowns in (3.3), we iteratively update each variable while fixing the others. The update rules are as follows

$$\mathcal{S}^{(i,k)} = \mathcal{L}^{(i)} \times_1 \left(\mathbf{U}^{(1,k)} \right)^T \dots \times_n \left(\mathbf{U}^{(n,k)} \right)^T, \quad (3.5)$$

$$\mathbf{U}^{(j,k)} = \mathbf{Z}^{(j,k)} \left(\left(\mathbf{Z}^{(j,k)} \right)^T \mathbf{Z}^{(j,k)} \right)^{-1/2}, \quad (3.6)$$

where

$$\mathbf{Z}^{(j,k)} = \sum_{i=1}^{N_l} \mathbf{M}_{i,k} \mathcal{L}_{(j)}^{(i)} \left(\mathbf{U}^{(n,k)} \otimes \dots \otimes \mathbf{U}^{(j+1,k)} \otimes \mathbf{U}^{(j-1,k)} \otimes \dots \otimes \mathbf{U}^{(1,k)} \right) \left(\mathcal{S}_{(j)}^{(i,k)} \right)^T. \quad (3.7)$$

Moreover,

$$\mathbf{M}_{i,k} = \left(\sum_{b=1}^K e^{\beta(\delta_k^i - \delta_b^i)} \right)^{-1}, \quad (3.8)$$

where

$$\delta_k^i = \left\| \mathcal{L}^{(i)} - \mathcal{S}^{(i,k)} \times_1 \mathbf{U}^{(1,k)} \cdots \times_n \mathbf{U}^{(n,k)} \right\|_F^2. \quad (3.9)$$

The temperature parameter, β , in equation (3.8) is initialized with a small value. For a fixed value of β , the sequential updates are repeated until the convergence of Ψ . Then, the temperature parameter is increased and the same update procedure is repeated. The algorithm converges when $\mathbf{M}$ is binary. The supplementary materials of Paper B contains a pseudo code describing the training process in more details. The derivation of the update rules above is a fairly straightforward extension of the methods used in [105] from 2D data points to general n D data points. Using the method of Lagrange multipliers, we can rewrite (3.3) as

$$\begin{aligned} & \sum_{i=1}^{N_l} \sum_{k=1}^K \mathbf{M}_{i,k} \left\| \mathcal{L}^{(i)} - \mathcal{S}^{(i,k)} \times_{j=1}^n \mathbf{U}^{(j,k)} \right\|_F^2 \\ & + \frac{1}{\beta} \sum_{i=1}^{N_l} \sum_{k=1}^K \mathbf{M}_{i,k} \log(\mathbf{M}_{i,k}) + \sum_{i=1}^{N_l} \mu_i \left(\sum_{k=1}^K \mathbf{M}_{i,k} - 1 \right) \\ & + \sum_{j=1}^n \sum_{k=1}^K \text{Tr} \left(\Lambda^{(j,k)} \left(\left(\mathbf{U}^{(j,k)} \right)^T \mathbf{U}^{(j,k)} - \mathbf{I} \right) \right), \quad (3.10) \end{aligned}$$

where μ_i and $\Lambda^{(j,k)}$ are Lagrange multipliers (note that $\Lambda^{(j,k)}$ are symmetric matrices). The term $\mathbf{M}_{i,k} \log(\mathbf{M}_{i,k})$ is an entropy barrier function [231] that ensures the positivity of $\mathbf{M}_{i,k}$. The update rule for $\mathbf{M}_{i,k}$ in (3.8) is independent of the data dimensionality as long as an error function associating a data point to a dictionary is defined (e.g. equation (3.9)). See [232] for the derivation of (3.8). Moreover, the update rule for $\mathcal{S}^{(i,k)}$ in (3.5), followed by the nullification of the small elements of $\mathcal{S}^{(i,k)}$, is a trivial extension of Theorem 2 in [105]. For the update rule of $\mathbf{U}^{(j,k)}$, $k \in \{1, \dots, K\}$, $j \in \{1, \dots, n\}$, we first gather and rewrite the terms in (3.10) that are dependent on $\mathbf{U}^{(j,k)}$ (i.e. one dictionary element). Using (2.9), and due to the invariance of the Frobenius norm to tensor unfolding, we can rewrite (3.10) as

$$\sum_{i,k} \mathbf{M}_{i,k} \text{Tr}(\mathbf{G}^T \mathbf{G}) + \sum_{k=1}^K \text{Tr} \left(\Lambda^{(j,k)} \left(\left(\mathbf{U}^{(j,k)} \right)^T \mathbf{U}^{(j,k)} - \mathbf{I} \right) \right) + C, \quad (3.11)$$

where C contains the terms that are independent of $\mathbf{U}^{(j,k)}$, and

$$\mathbf{G} = \mathcal{L}_{(j)}^{(i)} - \mathbf{U}^{(j,k)} \mathcal{S}_{(j)}^{(i,k)} \left(\bigotimes_{\alpha=\{n, \dots, 1\} \setminus j} \mathbf{U}^{(\alpha,k)} \right)^T. \quad (3.12)$$

For notational brevity, define

$$\bar{\mathbf{U}} \triangleq \bigotimes_{\alpha=\{n,\dots,1\}\setminus j} \mathbf{U}^{(\alpha,k)}. \quad (3.13)$$

Then, expanding the term $\text{Tr}(\mathbf{G}^T \mathbf{G})$ in (3.11), and using the properties of the matrix trace, we obtain

$$\begin{aligned} \sum_{i,k} \mathbf{M}_{i,k} & \left[\text{Tr} \left(\left(\boldsymbol{\mathcal{L}}_{(j)}^{(i)} \right)^T \boldsymbol{\mathcal{L}}_{(j)}^{(i)} \right) - 2 \text{Tr} \left(\left(\boldsymbol{\mathcal{L}}_{(j)}^{(i)} \right)^T \mathbf{U}^{(j,k)} \boldsymbol{\mathcal{S}}_{(j)}^{(i,k)} \bar{\mathbf{U}}^T \right) \right. \\ & \quad \left. + \text{Tr} \left(\bar{\mathbf{U}} \left(\boldsymbol{\mathcal{S}}_{(j)}^{(i,k)} \right)^T \left(\mathbf{U}^{(j,k)} \right)^T \mathbf{U}^{(j,k)} \boldsymbol{\mathcal{S}}_{(j)}^{(i,k)} \bar{\mathbf{U}}^T \right) \right] + \\ & \quad \sum_{k=1}^K \text{Tr} \left(\Lambda^{(j,k)} \left(\left(\mathbf{U}^{(j,k)} \right)^T \mathbf{U}^{(j,k)} - \mathbf{I} \right) \right) + C. \end{aligned} \quad (3.14)$$

Setting the derivatives with respect to $\mathbf{U}^{(j,k)}$ and $\Lambda^{(j,k)}$ to zero leads to the following system of equations (see [233] for a comprehensive review of matrix derivative rules)

$$\begin{cases} 2 \sum_i \mathbf{M}_{i,k} \left(-\boldsymbol{\mathcal{L}}_{(j)}^{(i)} + \mathbf{U}^{(j,k)} \boldsymbol{\mathcal{S}}_{(j)}^{(i,k)} \bar{\mathbf{U}}^T \right) \bar{\mathbf{U}} \left(\boldsymbol{\mathcal{S}}_{(j)}^{(i,k)} \right)^T + \mathbf{U}^{(j,k)} \left(\left(\Lambda^{(j,k)} \right)^T + \Lambda^{(j,k)} \right) = 0, \\ \left(\mathbf{U}^{(j,k)} \right)^T \mathbf{U}^{(j,k)} - \mathbf{I} = 0 \end{cases} \quad (3.15)$$

A similar optimization problem was considered in [216], where the authors propose a closed form solution for obtaining a basis given a collection of one-dimensional data points and subject to an orthonormality constraint. According to [216], the solution of (3.15) can be calculated as $\mathbf{U}^{(j,k)} = \mathbf{L} \mathbf{R}^T$, where $\mathbf{L}$ and $\mathbf{R}$ are left and right singular vectors of $\mathbf{Z}^{(j,k)}$, defined in (3.7).

3.2.3 Testing

Let $\{\boldsymbol{\mathcal{T}}^{(i)}\}_{i=1}^{N_t}$ denote a test set containing N_t data points, where the test set is a collection of data points that we would like to compress. The process of obtaining a set of sparse coefficient tensors corresponding to the data points in the test set is called *testing*. In addition to the sparse coefficient tensors, the testing process should also determine the most suitable dictionary in the MDE for each data point. Thanks to the orthonormality of the dictionaries, the testing process is simple. In Algorithm 2, we outline a greedy algorithm based on [105] that achieves this goal. The algorithm takes as input a data point and an MDE, and outputs a sparse coefficient tensor, as well as an integer holding the index of a dictionary in the MDE that leads to the *best* sparse representation of the data point. Therefore

Input: A data point $\mathcal{T}^{(i)}$ in the test set, error threshold ϵ , sparsity τ_t , and an MDE $\Psi = \{\mathbf{U}^{(1,k)}, \dots, \mathbf{U}^{(n,k)}\}_{k=1}^K$.

Result: A membership index, a , and a sparse tensor, $\mathcal{S}^{(i)}$.

Init.: $\mathbf{e} \in \mathbb{R}^K \leftarrow \infty$ and $\mathbf{z} \in \mathbb{R}^K \leftarrow \mathbf{1}$

```

1 for  $k = 1$  to  $K$  do
2    $\mathcal{X}^{(k)} \leftarrow \mathcal{T}^{(i)} \times_1 (\mathbf{U}^{(1,k)})^T \dots \times_n (\mathbf{U}^{(n,k)})^T$ ;
3   while  $\mathbf{z}_k \leq \tau_t$  and  $\mathbf{e}_k > \epsilon$  do
4     Nullify  $(\prod_{j=1}^n m_j) - \mathbf{z}_k$  smallest elements of  $\mathcal{X}^{(k)}$ ;
5      $\mathbf{e}_k \leftarrow \|\mathcal{T}^{(i)} - \mathcal{X}^{(k)} \times_1 \mathbf{U}^{(1,k)} \dots \times_n \mathbf{U}^{(n,k)}\|_F^2$ ;
6      $\mathbf{z}_k = \mathbf{z}_k + 1$ ;
7   end
8 end
9  $a \leftarrow \text{index of } \min(\mathbf{z})$ ;
10 if  $\mathbf{z}_a = \tau_t$  then  $a \leftarrow \text{index of } \min(\mathbf{e})$ ;
11  $\mathcal{S}^{(i)} \leftarrow \mathcal{X}^{(a)}$ ;
```

Algorithm 2: Compute coefficients and the membership index. The algorithm implements $\{\mathcal{S}^{(i)}, a\} = \text{Test}(\mathcal{T}^{(i)}, \Psi, \tau_t, \epsilon)$.

the algorithm is repeated for each data point in the test set. Clearly we prefer each data point to be represented by the smallest number of coefficients to achieve sparsity. However, if two or more dictionaries produce a representation with equal sparsity, then we choose the dictionary with the smallest reconstruction error. Two parameters are used to achieve this goal: a sparsity parameter, τ_t , and an error threshold, ϵ . Note that τ_t is merely an upper bound for sparsity and that we do not need an accurate estimate of τ_t . The threshold, ϵ , controls the amount of tolerable error in the sparse representation. As an example, if we set $\tau_t = \prod_{j=1}^n m_j$, i.e. the maximum possible value, then the amount of sparsity of a data point only depends on ϵ . In this way, we obtain coefficient tensors with variable sparsity over the test set.

We denote the process of obtaining the sparse coefficient tensors and a membership vector from a test set and an MDE by the following function

$$\left[\{\mathcal{S}^{(i)}\}_{i=1}^{N_t}, \mathbf{m} \right] = \text{Test} \left(\{\mathcal{T}^{(i)}\}_{i=1}^{N_t}, \Psi, \tau_t, \epsilon \right), \quad (3.16)$$

where the vector $\mathbf{m} \in \mathbb{R}^{N_t}$ contains the membership indices obtained from Algorithm 2 for all the data points in the test set; i.e. the value of $\mathbf{m}_i$ is an index pointing to

a dictionary in the MDE that is used for the data point $\mathcal{T}^{(i)}$. If the dictionaries are not orthonormal, then one has to solve an ℓ_0 problem using e.g. OMP, as discussed in Section 2.4. The computational complexity of Algorithm 2 is significantly lower than OMP, which is among the fastest ℓ_0 solvers. The speedup achieved using an MDE facilitates the compression of large-scale datasets. We will discuss the examples of compressing large-scale visual data in Chapter 4.

3.3 Pre-clustering

In this section, we discuss an algorithm for improving the performance of MDE with regards to the computational complexity of the training and the representativeness of the ensemble. We achieve this by performing a pre-clustering of the training set, followed by training an MDE for each pre-cluster. This approach assists the MDE training in finding a more suitable set of dictionaries for sparse representation. Our motivation for pre-clustering is described in more details in Section 3.3.1. Moreover, the pre-clustering algorithm will be presented in Section 3.3.2.

3.3.1 Motivation

The training algorithm discussed in Section 3.2 is a powerful tool for sparse representation of visual data. In Section 1.1 we discussed how sometimes there are data points that do not admit a sparse representation in a dictionary (see Fig. 1.3). Our solution was to add a new atom (in the case of an overcomplete dictionary), or to add a new orthogonal dictionary (for an orthogonal dictionary ensemble). Considering complex datasets such as light fields, it is not unlikely that a trained MDE with a limited number of dictionaries would fail in the sparse representation of the entire domain of light fields. Given that a test set containing all the possible light fields is practically infinite, we can only improve our model to alleviate the problem of representation error. One approach is to train an ensemble with a larger number of dictionaries. However, with large-scale datasets, training a large number of dictionaries quickly becomes intractable. In Paper B, we choose a different approach by applying *pre-clustering* to the training set, followed by learning an ensemble for each pre-cluster. We use the term “pre-clustering” since learning an MDE includes a clustering of the training set using the membership matrix $\mathbf{M}$.

An important factor in any clustering algorithm is the choice of a metric for computing pairwise distances between data points. For instance, K-Means [16] uses an ℓ_2 distance metric between points and have been used in computer graphics and visualization for different applications [228, 234, 235, 236, 237]. K-Means is also an important component in other clustering algorithms, e.g. spectral clustering [238, 239]. In Section 3.2 we described the requirements for choosing a dictionary in an ensemble. In particular, the metric is defined to be based on the sparsity

Table 3.3: Table of variables and parameters used for training an AMDE.

symbol	description	dimension
C	number of pre-clusters	scalar
$\mathbf{c}$	pre-cluster membership	$\mathbb{R}^{N_l}$
N_r	number of representatives	scalar
p	number of CP components	scalar
ϵ	error threshold	scalar

Input: The training set $\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}$, the number of clusters C , and the number of representative points N_r

Result: The pre-cluster membership vector $\mathbf{c} \in \mathbb{R}^{N_l}$

- 1 Rearrange $\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}$ into $\mathbf{F} \in \mathbb{R}^{N_l \times m_1 m_2 \dots m_n}$;
- 2 Compute $\mathbf{F} = \mathbf{A}\mathbf{B}^T$ using CP with p principal components;
- 3 Apply K-Means to rows of $\mathbf{A}$ with N_r clusters;
- 4 $\{\bar{\mathcal{L}}^{(i)}\}_{i=1}^{N_r} \leftarrow$ the centroids of $\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}$ using cluster indices returned by K-Means;
- 5 $\{\bar{\mathbf{U}}^{(1,c)}, \dots, \bar{\mathbf{U}}^{(n,c)}\}_{c=1}^C = \mathbf{Train}\left(\{\bar{\mathcal{L}}^{(i)}\}_{i=1}^{N_r}, C, \tau_l\right)$;
- 6 $\mathbf{c} = \mathbf{Test}\left(\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}, \{\bar{\mathbf{U}}^{(1,c)}, \dots, \bar{\mathbf{U}}^{(n,c)}\}_{c=1}^C, \tau_l, \epsilon\right)$;

Algorithm 3: The pre-clustering algorithm implementing $\mathbf{c} = \mathbf{Precluster}\left(\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}, C, N_r, p, \epsilon\right)$

and the representation error of a data point with respect to the dictionaries in the ensemble. Therefore, we would like the pre-clustering algorithm to use a similar metric for grouping the data points in the training set. Ideally, if the pre-clustering is successful, we obtain a set of pre-clusters where the data points in each pre-cluster are similar in terms of sparsity. In this way, the remaining variations in sparsity within a pre-cluster will be captured by the clustering done during training. From a different perspective, we are performing two layers of clustering prior to the training of the multidimensional dictionaries. Indeed one can perform additional levels of nested clustering of the training set, which is expected to perform better in capturing inter and intra coherences present in the data.

3.3.2 Aggregate Multidimensional Dictionary Ensemble

Paper B presents a pre-clustering algorithm that groups the data points based on their sparsity and representation error. In addition, the pre-clustering is designed such that the effect of noise and outliers in the data is reduced. For convenience, Table 3.3 includes all the parameters used for the training of a pre-clustered MDE. Algorithm 3 presents a pseudo code for the pre-clustering method, and in what follows we provide a brief description as well as our motivation for each of the steps involved.

First, we reduce the dimensionality of the data points in the training set. For notational brevity, define $m = \prod_{j=1}^n m_j$. The decomposition $\mathbf{F} = \mathbf{A}\mathbf{B}^T$ is performed, where $\mathbf{F} \in \mathbb{R}^{N_l \times m}$ contains vectorized data points as rows, $\mathbf{A} \in \mathbb{R}^{N_l \times p}$ contains coefficients in each row, and $\mathbf{B} \in \mathbb{R}^{m_1 m_2 \dots m_n \times p}$ has a set of basis vectors as columns. When $p \ll m$, we achieve a dimensionality reduction of the data points. In other words, the rows of $\mathbf{A}$ are now relatively low dimensional vectors that approximate the large multidimensional data points of the training set. A popular choice for performing this decomposition is Principal Component Analysis (PCA). However, PCA is known to be notoriously sensitive to outliers and noise [240]. Unfortunately, visual datasets are prone to noise and outliers due to the imperfections in the sensing process used to capture the data. Instead of PCA, we propose to use Coherence Pursuit (CP) [241], a Robust-PCA algorithm [211, 242] that is simple and fast, with strong theoretical guarantees.

Next, we apply K-Means on the rows of $\mathbf{A}$ with N_r clusters. Our goal here is to obtain a subset of the training set that is representative of the entire set. The user-defined parameter N_r controls the number of representative data points. Using the obtained cluster indices from K-Means, the centroids of $\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}$ are calculated. In other words, we calculate the mean of data points that are inside a K-Means cluster. One can also choose a data point from a K-Means cluster at random. We call these centroids the *representative set*, denoted $\{\bar{\mathcal{L}}^{(i)}\}_{i=1}^{N_r}$. The representative set is then used to train an MDE with C dictionaries, where C is the number of user-defined pre-clusters. Finally, the trained MDE, together with the entire training set, are used to produce a membership vector $\mathbf{c} \in \mathbb{R}^{N_l}$ using Algorithm 2, which associates each data point in the training set to a pre-cluster. We use the following function to denote the pre-clustering of a training set

$$\mathbf{c} = \text{Precluster} \left(\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}, C, N_r, p, \epsilon \right), \quad (3.17)$$

A thorough numerical analysis of the user-defined parameters (see Table 3.1) is presented in the supplementary materials of Paper B. The most important parameters are C and K . In practice, we would like to increase C and K as long as the computational cost of the training is acceptable. The values of these two

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

Result: An AMDE $\{\mathbf{U}^{(1,k)}, \dots, \mathbf{U}^{(n,k)}\}_{k=1}^{CK}$

```

1 c = Precluster  $\left(\{\mathcal{L}^{(i)}\}_{i=1}^{N_l}, C, N_r, p, \epsilon\right)$  using Algorithm 3;
2 for  $c = 1$  to  $C$  do
3    $\{\mathcal{X}^{(j)}\} \subset \{\mathcal{L}^{(i)}\}_{i=1}^{N_l}$  such that  $\mathbf{c}_j = c$ ;
4    $\{\mathbf{U}^{(1,k,c)}, \dots, \mathbf{U}^{(n,k,c)}\}_{k=1}^K = \mathbf{Train}(\{\mathcal{X}^{(j)}\}, K, \tau_l)$ ;
5 end
6 Compute the aggregated ensemble  $\Psi$  using (3.18);
```

Algorithm 4: Pre-clustered training to obtain an AMDE. This is performed only once for each application

parameters are proportional to the reconstruction quality because they decrease the representation error.

Since each MDE is a set, we combine all the MDEs that were trained for the pre-clusters into a single MDE. Formally,

$$\Psi = \bigcup_{c=1}^C \{\mathbf{U}^{(1,k,c)}, \dots, \mathbf{U}^{(n,k,c)}\}_{k=1}^K = \{\mathbf{U}^{(1,k)}, \dots, \mathbf{U}^{(n,k)}\}_{k=1}^{CK}. \quad (3.18)$$

With a slight abuse of notation, the new ensemble is also denoted by Ψ . We call the union of MDEs in 3.18 an *Aggregate Multidimensional Dictionary Ensemble (AMDE)*. To summarize, Algorithm 4 presents our approach for training an AMDE.

An important advantage of using an AMDE is that one can expand this set by adding more dictionaries. As an example, assume that we train an AMDE on a set of natural light fields. A direct application of this AMDE on synthetic light fields may lead to inferior results compared to an AMDE that is trained on synthetic datasets. In this case, one can simply combine the two trained AMDEs to form a new one; and Algorithm 2 will choose the best dictionary in the combined AMDE for a given data point (synthetic or natural). This property is not possessed by other approaches for dictionary learning (see Section 2.4). For instance, K-SVD cannot be expanded by the addition of new atoms, since a new training procedure has to be invoked.

3.4 Summary and Future Work

In this chapter, we presented an efficient learning based algorithm for sparse representation of visual data. The new method, i.e. AMDE, has several properties that distinguishes it from previous works. First, it enables sparse representation of multidimensional visual data since the algorithm is independent of dimensionality. As a result, any type of discrete visual data can be efficiently represented in this basis. Second, the pre-clustering algorithm, combined with the clustering of MDE, enables extremely sparse representations that are adapted to the given dataset. This means that the AMDE is more representative, as defined in Section 2.4, compared to an overcomplete or an analytical dictionary. Therefore, the AMDE shows better performance in handling visual datasets with large variations within the data, which will be confirmed in Chapter 4, where we examine the efficiency of AMDE in various applications. Third, AMDE enables real time rendering of compressed visual data. This property is achieved by our multidimensional representation, as well as the orthogonality of the dictionaries. Finally, compared to the widely used 1D dictionary training algorithms [162, 165, 172, 175, 192], the AMDE has orders of magnitude smaller memory footprint.

For future work, we believe that the extension of AMDE by applying nested levels of clustering is an interesting direction for research. Moreover, by extending orthonormal dictionary elements to overcomplete dictionary elements, one can achieve significant improvements on the representativeness of AMDE. However, this causes the reconstruction to become significantly slower, hence removing the real time performance of the AMDE. Note that Algorithm 2 is only valid for an orthonormal dictionary ensemble, and when we have an overcomplete ensemble, it is required to solve an ℓ_0 or ℓ_1 problem to obtain the sparse coefficients. While GPU-based implementations of such algorithms exist [243, 244, 245], our initial results show that several improvements are needed for utilizing them for large-scale visual datasets. Another interesting direction for improving the AMDE is to perform the pre-clustering and the MDE training stages in an iterative manner. In other words, we can use the trained AMDE to guide the pre-clustering in the next iteration (the first iteration of this method is Algorithm 4).

Compression of Visual Data

Compression has been utilized in computer graphics in several applications [97, 98, 246, 247, 248]. However, learning based methods for the sparse representation of visual data have received little attention. Hence, one of the main goals of this thesis is to describe and analyze methods for efficient sparse representations of visual data that enable high compression ratios with minimal degradation in image quality, as well as real time rendering of the compressed data. Utilizing the MDE and the AMDE, introduced in Chapter 3, in this chapter we will present three applications for the efficient compression of visual data using sparse representations. While a full compression framework may entail several other steps, such as entropy coding, we only focus on finding suitable transformations of the visual data using the learning based dictionary ensembles. Indeed any entropy coding algorithm can be applied on top of the methods we introduce in this chapter.

4.1 Outline and Contributions

In Section 4.2, we utilize MDE and K-Means to compress surface light fields for real time photorealistic rendering of highly complex static scenes. The method was presented in Paper A and the results show significant improvements over clustered principal component analysis [228], spherical harmonics [44], and K-SVD. In Section 4.3, we use the AMDE for the compression of light field and light field video datasets. The proposed method outperforms well-established analytical basis functions, as well as training based dictionaries. More importantly, we show that the AMDE enables real time GPU-based rendering of high resolution light field

videos that consume hundreds of gigabytes of storage. More detailed analysis of these results can be found in Paper B. Finally, in Section 4.4, we utilize the AMDE for the compression of datasets obtained from a device for appearance capture of relatively large scenes. A rotating arc containing several high resolution cameras covering the full hemisphere above the scene captures more than 300 gigabytes of data in a few seconds. After the compression of the data using the AMDE, real time exploration of the scene using e.g. virtual reality goggles is made possible. While the results are not published yet, they are included in this thesis to show the effectiveness of the AMDE in various applications. We would like to emphasize that since the AMDE can be trained on any discrete visual dataset with an arbitrary dimensionality, the results presented here are merely examples of possible applications for the proposed method.

4.2 Precomputed Photorealistic Rendering

This section will present a method for real time photorealistic rendering of highly complex scenes by pre-computing the Surface Light Field (SLF) of a scene. Given the large amount of data generated by SLFs, we apply a variant of MDE to compress this data. Moreover, we demonstrate high-quality real time rendering of the compressed SLF datasets using the GPU. We start by stating our motivation for choosing precomputed photorealistic rendering over approximate real time ray tracing methods in Section 4.2.1. An overview of the method is presented in Section 4.2.2. Moreover, various stages of the algorithm are described in sections 4.2.3–4.2.5, followed by the results in Section 4.2.6.

4.2.1 Background and Motivation

Photorealistic rendering is one of the key challenges in computer graphics. Decades of research has culminated in numerous methods for achieving photorealism in rendered images. While a significant progress has been made to this date, our definition of “photorealism” is also evolving. Moreover, as new techniques for rendering emerge, the amount of required computational power increases dramatically. Inadequate computing power has been a common trend in the field since the dawn of photorealistic rendering techniques [249, 250], a trend that is expected to continue in the future as our expectations of photorealism evolves. Today, a photorealistic rendering of a scene can take minutes to hours, or even days on multi-processor and multi-computer architectures. As a result, real time photorealistic rendering on a consumer level personal computer is a very challenging task. Several methods have been proposed in this direction, mostly relying on approximation techniques for ray tracing that utilize modern GPUs. However, existing methods only focus on certain components of the photorealistic rendering, or simulate a certain natural

Figure 4.1: Surface light field parameterization: **(a)** Illustration of HRDF matrices defined on a surface and **(b)** Spatial and angular sampling of the Stanford bunny.

phenomenon. For instance, several methods have been proposed for achieving real time photorealism of defocus blur [251], lens flare effects [252], layered materials [253], diffraction effects [254], rainbows [255], ocean surface [14], fur [256], and ray tracing of implicit surfaces [257] and isosurfaces [258]. Recently, Silvennoinen and Lehtinen [259] propose a real time global illumination technique with dynamic light sources in static scenes. However, the method is limited to diffuse materials only.

A different approach for real time photorealistic rendering is to pre-compute computationally intensive elements of the rendering pipeline. An early example of this approach was presented in [31], where a set of captured photos of a convex scene are resampled to an SLF. The SLF is then compressed using PCA, followed by vector quantization. The proposed method achieves real time performance for static scenes comprising of synthetic and scanned objects. Ren et al. [260] propose to learn a radiance regression function using neural networks that takes as input a position, view direction, and lighting conditions to calculate indirect illumination. The scene is assumed to be static, while the light sources can be dynamic. The main shortcoming of this approach is that only point light sources can be used. Moreover, the direct lighting is performed with an approximation method which significantly reduces photorealism. Note that the radiance regression function is trained and tested on the same scene, however with different light source locations and ray origins.

4.2.2 Overview

In contrast to the methods described in Section 4.2.1, Paper A presents an algorithm for real time photorealistic rendering of scenes with arbitrarily complex materials and light sources. The proposed method does not enforce any limitations on the complexity of the scene, however we require the scene to be static. To achieve real time performance, we precompute the full global illumination of each object in a scene and store the result in a set of SLFs. The proposed method consists of three stages: 1. SLF data generation (Section 4.2.3), 2. SLF compression (Section 4.2.4), and 3. real time rendering of the compressed SLF data (Section 4.2.5). In what follows, a brief description of each stage is presented. The results are summarized in Section 4.2.6.

4.2.3 Data Generation

An SLF is a 6D function of variables $(x, y, z, \phi, \theta, \lambda)$ that describes the outgoing radiance from a point (x, y, z) on a geometrical object along the direction (ϕ, θ) and with wavelength λ . As a first step, we need to discretize an SLF over each geometrical object in the scene. To facilitate the discretization of the spatial domain (x, y, z) , we perform uniform sampling of the texture space of each object, hence obtaining a set of points on the surface of each object. This is shown in Fig. 4.1b. For angular sampling, we use concentric mapping [261] to transform a regular 2D grid to a set of points on the unit hemisphere (see Fig. 4.1b); this mapping is invertible. Therefore, the outgoing radiance on the hemisphere can be stored in a matrix. We call this matrix the Hemispherical Radiance Distribution Function (HRDF), see Fig. 4.1a. For each object in the scene, we create a set of N HRDFs of size $m_1 \times m_2$, where these parameters are user defined and describe the required angular resolution.

At each of the N spatial location on an object, we calculate the outgoing radiance along the directions defined by the HRDF. To reduce Monte Carlo noise, for each outgoing direction in the HRDF, we use several rays in a small neighborhood of the outgoing direction and take the mean of the obtained radiance values. A clear advantage of our method is that any photorealistic rendering technique can be used for populating the SLF for each object in the scene. Moreover, there are no limitations on the type of materials or light sources since the outgoing radiance is calculated by an offline renderer.

4.2.4 Compression

Once we have the set of HRDFs for each object, K-Means is performed to cluster the HRDFs. Then for each cluster, we train a 2D MDE as described in Section 3.2. Afterwards, the coefficients and the membership vector are calculated using

Figure 4.2: Real time photorealistic rendering from compressed SLF datasets using an NVIDIA GTX 560 from ancient times with 1GB of memory.

Algorithm 2 and stored to the disk. It should be noted that here we perform a training for each object of the scene. In other words, the training and testing sets are the same and we have to store an MDE for each object. An extension of this method is to train an AMDE on a large dataset of SLFs from different scenes. In this way, it is only required to perform Algorithm 2 in order to obtain the sparse coefficients and the dictionary memberships for any new given SLF dataset to be compressed.

4.2.5 Real Time Rendering

During the real time rendering of the scene, from the intersection position of a camera ray with an object, we obtain the index $i \in \{1, \dots, N\}$ of an HRDF using the texture coordinates at the intersection point. From the direction of the camera ray, the index of an element inside the HRDF is calculated using inverse concentric

Figure 4.3: Insets from Fig. 4.2, comparing a reference rendering, MDE*, and CPCA. The reference rendering uses raw SLF data from an offline renderer.

mapping. Indeed, for each image pixel to be rendered, we only need a single value from an HRDF that corresponds to the view direction. Obtaining this value is straightforward due to the orthonormality of the dictionaries. Let $\mathbf{H}_{x_1, x_2}^{(i)}$ be an element at location (x_1, x_2) of the i th HRDF, then

$$\mathbf{H}_{x_1, x_2}^{(i)} = \sum_{z=1}^{\mathbf{z}_i} \mathbf{U}_{x_1, l_1^z}^{(1, \mathbf{m}_i)} \mathbf{S}_{l_1^z, l_2^z}^{(i)} \mathbf{U}_{x_2, l_2^z}^{(2, \mathbf{m}_i)}, \quad (4.1)$$

where $\mathbf{m}$ is the membership vector, $\mathbf{z}_i$ is the number of nonzero elements in $\mathbf{S}^{(i)}$, and (l_1^z, l_2^z) defines the location of nonzero values in $\mathbf{S}^{(i)}$.

4.2.6 Results

To evaluate the method, we construct three scenes as shown in Fig. 4.2. For data generation, we used the PBRT renderer [15] because it is an open-source software and we could embed our SLF data generator. The geometry, materials, and light sources were designed to challenge the compression method rather than producing very realistic renderings. For instance, we use multiple points light sources and highly specular measured materials (e.g. gold and silver) to produce extremely high frequency variations inside each HRDF. Indeed this is very challenging for a compression method. The Cornell box in Fig. 4.2a contains an area light source on the ceiling and three point light sources. The scene in Fig. 4.2b contains 10 point light sources, and Fig. 4.2d shows our results for the same scene under image based lighting. Figure 4.2c has 6 point light sources and the gold ring has a perfect specular component to cast caustics on the plane. In Fig. 4.3 we include insets from these scenes and compare the results with CPCA compression, and the raw SLF data. Our method significantly outperforms CPCA in retaining the details of the SLF data. For very high frequency materials, e.g. the gold ring in Fig. 4.2c, CPCA fails to capture the details.

4.3 Light Field and Light Field Video

In Paper B, we use the AMDE for the compression of light fields and light field videos. For light fields, the Stanford Lego Gantry dataset¹ is used, which consists of twelve light fields with angular resolution of 17×17 and a variable spatial resolution (typically 1024×1024). We used 8×8 central views of the light fields and construct data points with dimensionality $m_1 = 5$, $m_2 = 5$, $m_3 = 8$, $m_4 = 8$, and $m_5 = 3$; i.e. we create 5×5 patches over each light field view image. The following parameters were used to train an AMDE: $C = 16$, $K = 8$, $\tau_l = 20$, $p = 128$, and $N_r = 4000$. Therefore, the dictionaries in the AMDE have the dimensionality

$$\left\{ \mathbf{U}^{(1,k)} \in \mathbb{R}^{5 \times 5}, \mathbf{U}^{(2,k)} \in \mathbb{R}^{5 \times 5}, \mathbf{U}^{(3,k)} \in \mathbb{R}^{8 \times 8}, \right. \\ \left. \mathbf{U}^{(4,k)} \in \mathbb{R}^{8 \times 8}, \mathbf{U}^{(5,k)} \in \mathbb{R}^{3 \times 3} \right\}_{k=1}^{128}. \quad (4.2)$$

¹ <http://lightfield.stanford.edu>

Figure 4.4: Visual quality comparison for a natural light field from the Stanford Lego Gantry dataset. The size of the light field is 240MB. False color insets are included to facilitate comparison.

The cover of this thesis illustrates the sparse representation of the *Tarot Cards* light field using an ensemble of two dictionaries with the same dimensionality as in (4.2).

Reconstruction results for the *Bracelet* light field are summarized in Fig. 4.4, where we compare to K-SVD, 5D Discrete Cosine Transform (DCT) [263], HOSVD, and CDF 9/7 (the basis used in JPEG2000). Apart from the three light fields we used for testing (*Bracelet*, *Lego Knights*, and *Tarot Cards*), the rest of the dataset was

Figure 4.5: Visual quality comparison of different compression methods for the *Trains* natural light field video from [262]. The parameters for all the methods were chosen so that the storage cost is the same. The dictionary is not included in storage cost.

used for training. AMDE is vastly superior compared to the analytical dictionaries and the tensor decomposition based methods. Moreover, compared to K-SVD, we see a large improvement in quality, as well as the time required for encoding and decoding (see Paper B). More importantly, the size of the K-SVD dictionary is 88MB, while AMDE is 0.046MB. Note that we did not include the size of the K-SVD dictionary in the calculation of its storage cost. And yet AMDE shows a

much superior image quality compared to K-SVD.

For light field videos, we used the dataset described in [262], which was captured using a 4×4 camera array at 30 frames per second. Each camera has a resolution of 2048×1088 . From this dataset, we create data points of size $m_1 = 10$, $m_2 = 10$, $m_3 = 4$, $m_4 = 4$, $m_5 = 3$, and $m_6 = 4$; i.e. the spatial patches are 10×10 and each data point contains four consecutive frames. Note that we have included all the angular information in each data point to have seamless angular reconstruction during the rendering. For training, we set the following parameters: $C = 4$, $K = 32$, $\tau_l = 100$, $p = 400$, and $N_r = 6000$. The training of AMDE was done on the *Painter* dataset to compress the *Trains* dataset, as shown in Fig. 4.5. These two light field videos are visually very different from each other. Moreover, unlike Lytro Illum light fields (see Fig. 6.5) which have a disparity of 1 to 5 pixels, the disparity of the *Trains* light field video is up to 150 pixels. Since there is little coherence along the angular dimension of each data point, the dataset we use here is very challenging for compression. Here, we also see the clear advantage of the AMDE, achieving about 2.0dB over K-SVD, even though the size of the K-SVD dictionary is 432MB and the size of the AMDE is 0.0627MB.

Reconstruction of light fields and light field videos is fairly straightforward. We state the reconstruction procedure for a light field video. Indeed when rendering a light field video, we only need to reconstruct a single light field view in a frame. Let $\mathcal{T}_{x_1, x_2, \dots, x_6}^{(i)}$ be an element of $\mathcal{T}^{(i)}$ that we would like to reconstruct. Since the dictionaries in the ensemble are orthonormal and the coefficients are sparse, we can use the following formula to obtain $\mathcal{T}_{x_1, x_2, \dots, x_6}^{(i)}$:

$$\mathcal{T}_{x_1, x_2, \dots, x_6}^{(i)} = \sum_{z=1}^{\mathbf{z}_i} \mathcal{S}_{l_1^z, l_2^z, \dots, l_6^z}^{(i)} \mathbf{U}_{x_1, l_1^z}^{(1, \mathbf{m}_i)} \mathbf{U}_{x_2, l_2^z}^{(2, \mathbf{m}_i)} \dots \mathbf{U}_{x_6, l_6^z}^{(6, \mathbf{m}_i)}, \quad (4.3)$$

where $\mathbf{m}$ and $\mathcal{S}^{(i)}$ are obtained from Algorithm 2, $\mathbf{z}_i$ is the number of nonzero elements in $\mathcal{S}^{(i)}$, and $(l_1^z, l_2^z, \dots, l_6^z)$ defines the location of nonzero values in $\mathcal{S}^{(i)}$.

Since the computational complexity of (4.3) is only related to the sparsity of data points, an efficient GPU-based implementation is indeed feasible. We developed an application for real time playback of high resolution light field videos, achieving more than 40 frames per second for the dataset of [262] and 200 frames per second for the Stanford Lego Gantry dataset using an NVIDIA Titan Xp GPU. The implementation utilizes multi-threading and pixel buffer objects (PBO) to accelerate the data transfer between the CPU and the GPU, hence enabling real time playback of lengthy light field videos on a consumer level PC.

(a) The large-scale appearance capture device. *Per Larsson* for scale, the man behind making the research gadgets at the Visual Computing Laboratory.

(b) Diagram of the capturing mechanism using 13 cameras.

Figure 4.6: SLF compression and rendering.

4.4 Large-scale Appearance Data

At the Visual Computing Lab (VCL), where the author of this thesis has been working as a PhD candidate, we have recently developed an appearance capture device for relatively large scenes, see Fig. 4.6a. The device consists of a circular frame 3.2 meters in diameter, mounted on a precision guided motor. A diagram of the device is shown in Fig. 4.6b. Several cameras and individually addressable LED light sources can be mounted on the frame. By rotating the frame that holds the cameras, the full outgoing radiance of a scene can be captured. Moreover, the LED light sources can be used for projecting special lighting patterns on the scene for e.g. estimating the geometry [264] or capturing the light transport of the scene [86]. The setup used for the experiments conducted for this thesis consists of 13 cameras with a resolution of 4112×3008 that are mounted on one side of the frame, as shown in Fig. 4.6b. A number of objects, as well as scenes containing multiple objects, have been captured using this device, where a capture results in 323.46 gigabytes of data when stored using half-precision floating point values [265].

In this section we report compression results using an AMDE for the two data sets demonstrated in Fig. 4.7. The azimuthal movements of the cameras are one degree apart, giving us 360 sets of images, where each set contains 13 images from 13 cameras along the elevation angle. Because the distance between two neighbouring cameras is relatively large, there is little coherence between the

Figure 4.7: Examples of cropped images captured using the device in Fig. 4.6

Figure 4.8: Compression results for the *Miffy* and *Drill* scenes captured using the device in Fig. 4.6. The PSNR is calculated over the entire dataset.

images of neighbouring cameras. As a result, we create data points from 360 images captured from each camera. The patch size over each image is set to 40×30 . Therefore, each data point has dimensions $m_1 = 40$, $m_2 = 30$, $m_3 = 3$, and $m_4 = 6$, where m_4 represents the number of consecutive azimuthal images that are included in a data point. Note that in contrast to the surface light field application described

in Section 4.2, here we do not have the geometry of the objects in the scene. The appearance data discussed in this section can be interpreted as spherical light fields [266].

In Section 3.3 we discussed an important advantage of AMDE in terms of flexibility. In particular, one can train several AMDEs from different datasets and take the union of them to obtain a more representative ensemble. Due to the fact that the images from different cameras are considerably distinct, we train three AMDEs for images captured using three cameras (from top: cameras 2, 7, and 13). Each AMDE was trained using the following parameters: $C' = 4$, $K = 8$, $\tau_l = 32$, $p = 128$, and $N_r = 5000$. Therefore, the final AMDE contains 96 dictionaries. After applying Algorithm 2 using the raw dataset and the AMDE, we store the coefficients to disk. Preliminary image quality results are included in Fig. 4.8. The compression ratio for the two datasets, without entropy coding of the sparse coefficients, is about 34 : 1. Moreover, a real time renderer is developed that enables exploration of the scene using e.g. virtual reality and head mounted displays.

4.5 Summary and Future Work

We applied the AMDE for compression of four different datasets: surface light fields (5D unfolded to 2D), light fields (5D), light field videos (6D), and appearance data (4D). A compression ratio of 10 : 1 up to 40 : 1 can be achieved even without applying entropy coding on the sparse coefficients. Given the flexibility of AMDE, we are interested in applying the proposed compression method to a variety of new datasets. In particular, an interesting direction for future work is the application of AMDE to BTF (7D or 8D), BRDF (5D), and SVBRDF (7D) datasets. Regarding the appearance capturing device discussed in Section 4.4, an area of future research is to utilize LED arrays for projecting dictionaries in order to directly measure the coefficients. Additionally, since AMDE has a small representation error, we believe that the sparse coefficients, as well as the membership vector, can be used in classification and recognition problems. Since the AMDE already performs a clustering, the application of Algorithm 2 leads to the classification of input data points. However, this classification is based on sparsity and error. Therefore, one needs to alter the problem definition in (3.3) by adding a constraint to enforce the dictionaries to be *discriminative* according to a predefined metric.

Another interesting area for future research is to design real time compression techniques. Initial steps in this direction has been taken. For instance, in [8] we present a GPU-based implementation of Algorithm 2 for fast encoding of light fields. Further experiments also show that real time compression of high resolution light field videos is certainly feasible. Such a technique has a broad range of applications, e.g. for real time capturing and storage of appearance data that we

discussed in Section 4.4. While compressed sensing, which is the topic of chapters 5 and 6, is a more efficient method to accelerate the capturing of visual data, it requires different sensing mechanisms, and hence new hardware implementations. In contrast, real time compression during the capturing can be applied on top of existing measurement devices for obtaining compact representations of BRDFs, BTFs, light field videos, and other types of visual data.

Compressed Sensing

For decades, the Shannon-Nyquist theorem [69, 70] defined the requirements for sampling signals. It states that a band-limited signal with no frequencies higher than f can be completely determined by placing samples at uniform (equally spaced) points that are $1/2f$ seconds apart, which implies sampling at a rate of $2f$ (known as the Nyquist rate). Uniform sampling exerts several limitations in practice, especially when sampling rate is high. The introduction of nonuniform sampling [267] opened up many new research directions. The Shannon-Nyquist theorem holds for nonuniform sampling as long as the average sampling rate exceeds the Nyquist rate [268]. This requirement can be relaxed

when we know the frequency band locations and their widths [269, 270], or using periodic non-uniform sampling [271], as well as reconstruction from local averages [272]. However, reconstruction methods used for uniform sampling do not apply to nonuniform sampling (see [273] for a review of recovery methods from nonuniform samples). Two important applications that deal with the problem of reconstructing a signal from its nonuniform frequency-domain samples are Computed Tomography

Figure 5.1: Saving time with mathematics¹.

¹ Image source: people.eecs.berkeley.edu/~mlustig/

(CT) and Magnetic Resonance Imaging (MRI) [274].

Compressed Sensing sparked a paradigm shift in signal sampling. The main results in compressed sensing state that a sparse signal can be exactly recovered with a sampling rate well-below the Nyquist rate. While sparse representation was a very active field in 1990s and 2000s for denoising and solving inverse problems using wavelets, mathematical formulation of conditions for exact recovery was never considered [122]. Just like many other ground-breaking inventions, compressed sensing was discovered by chance. In 2004, Emmanuel Candès was trying to reconstruct a sub-sampled Shepp-Logan Phantom image [275]. By his surprise, the result of the ℓ_1 problem turned out to be an exact recovery of the original image even though the sampling rate was well-below what he thought was the required number of measurements. This motivated him to find the reason behind this “magical” outcome and to formulate it. With the help of one of the greatest mathematicians of our time, Terrence Tao, they published some of the seminal works in compressed sensing [75, 76, 77]. In parallel, David Donoho presented an information theoretic perspective on recovery of sparse signals from only a few measurements [42], and for the first time dubbed the method “compressed sensing”.

The majority of the work presented in this thesis involves compressed sensing. Moreover, contributions of this thesis on the topic are both theoretical and empirical. This chapter will discuss the theoretical findings of this thesis, in which an outline of the materials to be discussed, as well as the contributions, are summarized in Section 5.1. In Chapter 6, we will present the empirical contributions of this thesis for utilizing compressed sensing in various computer graphics and imaging applications.

5.1 Outline and Contributions

Compressed sensing is indeed of paradoxical nature – exact reconstruction from what appears to be highly inadequate measurements. Therefore, rigorous analysis of the conditions for exact recovery is crucial. Before presenting the main theoretical contributions of this thesis regarding compressed sensing, key definitions that are used throughout this chapter are presented in Section 5.2. Section 5.3 formulates the compressed sensing problem for 1D signals, which is the most common way of formulating the problem, as well as n D signals. In addition, we discuss different types of measurement matrices in Section 5.4, which play an important role in deriving conditions for exact recovery. There are two families of theoretical analysis in compressed sensing. Universal theoretical results (Section 5.5) demonstrate the conditions for exact recovery by analyzing the properties of a measurement matrix, sparsity, and the number of measurements. There exists a large body of research on deriving metrics for the suitability of a measurement matrix with regards to

exact recovery. We will review a number of these metrics in sections 5.5.1–5.5.4 and present some of the key results regarding the universal recovery conditions for sparse signals.

Aside from universal results, there exists a family of theoretical results on the convergence of specific algorithms solving the compressed sensing problem (e.g. using ℓ_0 , ℓ_1 , or ℓ_p norms for $0 < p < 1$). Indeed these algorithm-specific results also consider sparsity, number of measurements, and the metrics on measurement matrices for exact recovery. However, these properties are considered in the context of the algorithm at hand. Algorithm-specific analyses typically perform better for an specific algorithm than universal results; however this comes at the cost of reducing the generality of the obtained theoretical results. This thesis makes contributions by presenting a novel algorithm-specific analysis, as well as a universal result for recovery of signals that are sparse under a dictionary ensemble.

One of the key contributions of this thesis is a theoretical analysis of Orthogonal Matching Pursuit (OMP), an ℓ_0 recovery method that is widely used and provides a good trade off between accuracy and computational cost. In Section 5.6, we describe the OMP algorithm and present its advantages and disadvantages from a practical point of view. Next, in Section 5.7, a theoretical analysis of the method is presented, where we derive conditions under which the OMP algorithm exactly recovers a sparse signal. This analysis is a combination of the theoretical results reported in Paper F and Paper G.

In Section 5.8, we present a theoretical analysis on the uniqueness of the solution obtained from an ℓ_0 problem when the signal is sparse in an ensemble of dictionaries (see Section 3.2). This analysis is based on a spike ensemble for point sampling, which has several applications in graphics and imaging. The theoretical results for an ensemble of 2D dictionaries (orthogonal or overcomplete) were reported in Paper D. Moreover, a generalized analysis for an ensemble of n D dictionaries was presented in Paper B. Finally, in Section 5.9 we summarize the contributions and present possible venues for future work.

5.2 Definitions

In what follows, a number of key definitions that will be used in the upcoming sections of this chapter are listed.

Definition 5.1 (signal). *A signal, in the context of the materials presented in this thesis, is a real or complex valued vector ($\mathbf{x} \in \mathbb{R}^m$), matrix ($\mathbf{X} \in \mathbb{R}^{m_1 \times m_2}$) or tensor ($\mathcal{X} \in \mathbb{R}^{m_1 \times m_2 \times \dots \times m_n}$).*

Formulations in this chapter are based on real-valued signals, unless stated otherwise.

Definition 5.2 (τ -sparse). A signal $\mathbf{x} \in \mathbb{R}^m$ is called τ -sparse under a dictionary $\mathbf{D} \in \mathbb{R}^{m \times k}$ if $\mathbf{x} = \mathbf{D}\mathbf{s}$ and $\|\mathbf{s}\|_0 = \tau$. Note that $\tau \leq m$, otherwise the representation is not unique. This definition can also be used for $\text{vec}(\mathbf{X})$ and $\text{vec}(\mathcal{X})$.

Definition 5.3 ($(\tau_1, \dots, \tau_n)$ -sparse). A signal $\mathcal{X} \in \mathbb{R}^{m_1 \times \dots \times m_n}$ is called $(\tau_1, \dots, \tau_n)$ -sparse under an nD dictionary $\{\mathbf{D}^{(i)} \in \mathbb{R}^{m_i \times k_i}\}_{i=1}^n$ if $\mathcal{X} = \mathcal{S} \times_1 \mathbf{D}^{(1)} \times_2 \dots \times_n \mathbf{D}^{(n)}$ and

$$\max_{\forall j \in \{1, \dots, \prod_{k=1, k \neq j}^n m_k\}} \|(\mathcal{S}^{(i)})_{:,j}\|_0 = \tau_i \quad (5.1)$$

Note that $\tau_i \leq m_i$, $\forall i \in \{1, \dots, n\}$, otherwise the uniqueness of representation cannot be guaranteed.

Definition 5.4 (support). Let $\mathbf{x} \in \mathbb{R}^m$ be τ -sparse under a dictionary $\mathbf{D} \in \mathbb{R}^{m \times k}$. The support of $\mathbf{x}$, denoted $\text{supp}(\mathbf{x})_{\mathbf{D}}$, is a set of natural numbers holding the location of nonzero values in $\mathbf{s}$. When it is clear from the context, we may drop the subscript, i.e. $\text{supp}(\mathbf{x})$. For matrices and tensors, the elements of the support set are tuples, e.g. $\{(i_1, i_2), (j_1, j_2), \dots\}$ and $\{(i_1, \dots, i_n), (j_1, \dots, j_n), \dots\}$, respectively.

Definition 5.5 (mutual coherence). The mutual coherence of a matrix $\mathbf{X} \in \mathbb{R}^{m \times k}$ is

$$\mu(\mathbf{X}) = \max_{1 \leq i \neq j \leq k} \frac{|\mathbf{X}_i^T \mathbf{X}_j|}{\|\mathbf{X}_i\|_2 \|\mathbf{X}_j\|_2}. \quad (5.2)$$

Definition 5.6 ((p, q) operator norm). The (p, q) operator norm of a matrix $\mathbf{X}$ is defined as

$$\|\mathbf{X}\|_{p,q} = \max_{\mathbf{b} \neq \mathbf{0}} \frac{\|\mathbf{X}\mathbf{b}\|_q}{\|\mathbf{b}\|_p} \quad (5.3)$$

5.3 Problem Formulation

In Section 1.2, we briefly introduced compressed sensing and formulated the problem of sampling a 1D sparse signal. For convenience, we include equations (1.1) and (1.2) below

$$\min_{\hat{\mathbf{s}}} \|\hat{\mathbf{s}}\|_0 \quad \text{s.t.} \quad \|\mathbf{y} - \Phi \mathbf{D} \hat{\mathbf{s}}\|_2^2 \leq \epsilon, \quad (5.4)$$

$$\min_{\hat{\mathbf{s}}} \|\hat{\mathbf{s}}\|_1 \quad \text{s.t.} \quad \|\mathbf{y} - \Phi \mathbf{D} \hat{\mathbf{s}}\|_2^2 \leq \epsilon, \quad (5.5)$$

where $\Phi \in \mathbb{R}^{s \times m}$ is a *measurement* or *sensing* matrix and $\mathbf{D} \in \mathbb{R}^{m \times k}$ is a dictionary that enables the sparse representation of an unknown signal $\mathbf{x} = \mathbf{D}\mathbf{s}$, $\|\mathbf{s}\|_0 = \tau$. The measurement model is formulated as $\mathbf{y} = \Phi \mathbf{x} + \mathbf{w}$, where $\mathbf{w}$ is the measurement

noise. The sensing matrix Φ maps a vector of dimension m to a vector of dimension s . The variable s denotes the *number of samples*, and is typically user-defined, unless the measurement matrix has a predefined size. If the measurements are assumed to be noise-free, i.e. $\mathbf{w} = \mathbf{0}$, then the constraints of (5.4) and (5.5) are replaced with $\mathbf{y} = \Phi \mathbf{D} \hat{\mathbf{s}}$. It is a common practice to define $\mathbf{A} = \Phi \mathbf{D}$ for convenience. The matrix $\mathbf{A}$ is called an *equivalent* measurement matrix.

The recovery of an n D signal, $\mathcal{X} \in \mathbb{R}^{m_1 \times \dots \times m_n}$, that is $(\tau_1, \dots, \tau_n)$ -sparse under an n D dictionary $\{\mathbf{U}^{(1)} \in \mathbb{R}^{m_1 \times k_1}, \dots, \mathbf{U}^{(n)} \in \mathbb{R}^{m_n \times k_n}\}$ is formulated as follows

$$\min_{\hat{\mathbf{s}}} \|\hat{\mathbf{s}}\|_0 \quad \text{s.t.} \quad \|\mathbf{y} - \hat{\mathbf{s}} \times_1 \Phi^{(1)} \mathbf{U}^{(1)} \times_2 \dots \times_n \Phi^{(n)} \mathbf{U}^{(n)}\|_2^2 \leq \epsilon, \quad (5.6)$$

where $\{\Phi^{(1)} \in \mathbb{R}^{s_1 \times m_1}, \dots, \Phi^{(n)} \in \mathbb{R}^{s_n \times m_n}\}$ are measurement matrices that operate *independently* along each dimension, and $\{s_1, \dots, s_n\}$ are the number of samples along each dimension. Moreover, $\mathbf{y} = \mathcal{X} \times_1 \Phi^{(1)} \times_2 \dots \times_n \Phi^{(n)} + \mathbf{w}$ defines the measurement model. It is possible to convert (5.6) to an equivalent 1D form with independent measurement matrices

$$\min_{\hat{\mathbf{s}}} \|\hat{\mathbf{s}}\|_0 \quad \text{s.t.} \quad \|\mathbf{y} - (\Phi^{(n)} \mathbf{U}^{(n)} \otimes \dots \otimes \Phi^{(1)} \mathbf{U}^{(1)}) \hat{\mathbf{s}}\|_2^2, \quad (5.7)$$

where $\mathbf{y} = (\Phi^{(n)} \otimes \dots \otimes \Phi^{(1)}) \text{vec}(\mathcal{X})$. Another possibility is to convert (5.6) to an equivalent 1D form, however with a single measurement matrix

$$\min_{\hat{\mathbf{s}}} \|\hat{\mathbf{s}}\|_0 \quad \text{s.t.} \quad \|\mathbf{y} - \Phi (\mathbf{U}^{(n)} \otimes \dots \otimes \mathbf{U}^{(1)}) \hat{\mathbf{s}}\|_2^2 \leq \epsilon, \quad (5.8)$$

where $\mathbf{y} = \Phi \text{vec}(\mathcal{X})$ and $\Phi \in \mathbb{R}^{s \times (m_1 m_2 \dots m_n)}$. An advantage of (5.8) is that the number of samples, s , can be fine tuned, whereas in (5.6) and (5.7) the total number of samples is $s = s_1 s_2 \dots s_n$.

5.4 The Measurement Matrix

There exists numerous choices for populating a measurement matrix, denoted Φ . Two major categories exist in this regard: *random* and *deterministic*. Perhaps the most commonly used random measurement matrix is one with i.i.d. Gaussian entries. These matrices are provably optimal for exact recovery (see Section 5.5.4 for more details). Moreover, Gaussian matrices simplify the analysis of exact recovery conditions in various cases. Other types of random measurement matrices also exist. For instance, we can set the elements of Φ to $\{0, 1\}$ with probability 1/2 (i.e. the symmetric Bernoulli distribution), or to ± 1 with probability 1/2. Moreover, a measurement matrix can be constructed by sampling from the uniform distribution in the interval $[0, 1]$, which is essentially equivalent to sampling on the unit sphere

of $\mathbb{R}^s$. For random sensing matrices, the measurement model $\mathbf{y} = \Phi \mathbf{x} + \mathbf{w}$ dictates that one has to have access to all the elements of $\mathbf{x}$. However, in many applications, we would like to directly sample a subset of elements in $\mathbf{x}$, which is known as point sampling. For discrete signals, point sampling is modeled by a matrix, defined below.

Definition 5.7 (spike ensemble). Let Γ be the uniform random permutation of the set $\{1, \dots, m\}$. Define Λ as the first $s \leq m$ entries of Γ . Then the spike ensemble is defined as $\mathbf{I}_{\Lambda, \cdot}$, where $\mathbf{I} \in \mathbb{R}^{m \times m}$ is the identity matrix.

A spike ensemble has a nonzero element in each row. If there exists more than one nonzero element in a row, we obtain a *binary* sensing matrix [276].

Deterministic measurement matrices are categorized into *analytical* and *learning based*. Analytical matrices are constructed using mathematical tools such that strong theoretical guarantees can be achieved for the solution of (5.4) or (5.5); a few examples include [277, 278, 279, 280, 281, 282, 283]. On the other hand, learning based methods use an optimization problem to generate a matrix Φ such that the equivalent measurement matrix $\mathbf{A} = \Phi \mathbf{D}$ produces a more accurate solution to (5.4) or (5.5). A few examples in this category include [284, 285, 286, 287].

There exists an additional family of sensing matrices that have deterministic structures but may contain random elements. For instance, random circulant and Toeplitz matrices have been used for compressed sensing [288, 289, 290, 291, 292]. Moreover, the measurement matrix for a single sensor light field camera is block-circulant with random elements. This type of sensing matrix will be described in detail in Section 6.2.

5.5 Universal Conditions for Exact Recovery

Universal results in compressed sensing state the required conditions for recovering a sparse signal from incomplete measurements, *regardless* of the algorithm used for solving the problem. Formulating universal results requires the assumption that the recovery algorithm converges to a solution. While this is true for the ℓ_1 formulation (i.e. (5.5)), there is no guarantee that an ℓ_0 recovery may lead to the most sparse solution, and that the solution is unique. Hence, universal results typically state the equivalence conditions for the solutions of ℓ_0 and ℓ_1 problems.

5.5.1 Uncertainty Principle and Coherence

Heisenberg's uncertainty principle states that complementary physical properties of a particle, e.g. position and momentum wave functions, cannot be measured with arbitrary precision simultaneously. In mathematical terms, let $f(x)$ be any

absolutely continuous function such that $xf(x)$ and $f'(x)$ are square integrable [293]. Moreover, let $F(\omega)$ be the Fourier transform of $f(x)$. Uncertainty principle states that $f(x)$ and $F(\omega)$ obey the following inequality

$$\left(\int_{-\infty}^{\infty} x^2 |f(x)|^2 dx \right) \left(\int_{-\infty}^{\infty} \omega^2 |F(\omega)|^2 d\omega \right) \geq C, \quad (5.9)$$

where C is a constant. It can be noted that the integrals above calculate the second moments of $|f(x)|^2$ and $|F(\omega)|^2$ about zero. This is a well-known property in the classical sampling literature. For example, a box function in spatial domain is a Sinc function in the frequency domain and an impulse train (i.e. point samples) with period T in the spatial domain becomes an impulse train with period $1/T$ in the frequency domain.

Let us now translate the discussion above to the language of compressed sensing. Assume that we have a complex-valued signal $\mathbf{x} \in \mathbb{C}^m$. Let $\mathbf{I} \in \mathbb{R}^{m \times m}$ be the identity matrix and $\mathbf{I}_\alpha$ be a subset of columns in $\mathbf{I}$. Note that $\mathbf{I}_\alpha$ is a point sampling matrix that extracts the elements of $\mathbf{x}$ corresponding to α . Moreover, let $\mathbf{F} \in \mathbb{C}^{m \times m}$ be the Fourier transform matrix, and let $\mathbf{F}_\beta$ be a subset of columns in $\mathbf{F}$. Since both $\mathbf{I}$ and $\mathbf{F}$ are orthonormal bases, there exists $\mathbf{s}^{(1)}$ and $\mathbf{s}^{(2)}$ such that $\mathbf{x} = \mathbf{I}\mathbf{s}^{(1)} = \mathbf{F}\mathbf{s}^{(2)}$. In this setup, the uncertainty principle described in [294] shows that the inequality $|\alpha| + |\beta| \geq 2\sqrt{m}$ holds. In other words, $\mathbf{x}$ cannot be sparse in the canonical basis *and* the Fourier basis at the same time.

We can also write a sparse representation of $\mathbf{x}$ as

$$\mathbf{x} = [\mathbf{I}, \mathbf{F}] \underbrace{\begin{bmatrix} \mathbf{s}^{(1)} \\ \mathbf{s}^{(2)} \end{bmatrix}}_{\mathbf{s}}, \quad (5.10)$$

where $[\mathbf{I}, \mathbf{F}]$ is the concatenation of $\mathbf{I}$ and $\mathbf{F}$ along columns. We now have arrived at an underdetermined system of linear equation and several questions arise. For instance, is it possible to obtain a solution from

$$\min_{\hat{\mathbf{s}}} \|\hat{\mathbf{s}}\|_0 \quad \text{s.t.} \quad \mathbf{x} = [\mathbf{I}, \mathbf{F}]\hat{\mathbf{s}}, \quad (5.11)$$

or

$$\min_{\hat{\mathbf{s}}} \|\hat{\mathbf{s}}\|_1 \quad \text{s.t.} \quad \mathbf{x} = [\mathbf{I}, \mathbf{F}]\hat{\mathbf{s}}, \quad (5.12)$$

that is unique (i.e. the global minimizer)? And what is the relation between $|\alpha|$ and $|\beta|$ in the obtained unique solution? These questions were answered by Donoho and Hue in [295], which is summarize in the following theorem

Figure 5.2: An illustration of bounds stated in Theorem 5.3

Theorem 5.1 ([295]). Let $\alpha \subset \{1, \dots, m\}$ and $\beta \subset \{1, \dots, m\}$ be two index sets. Let $\mathbf{x} \in \mathbb{C}^m$ be defined by

$$\mathbf{x} = [\mathbf{I}_\alpha, \mathbf{F}_\beta] \begin{bmatrix} \mathbf{s}_\alpha^{(1)} \\ \mathbf{s}_\beta^{(2)} \end{bmatrix}, \quad (5.13)$$

where $\mathbf{I} \in \mathbb{R}^{m \times m}$ is the identity matrix and $\mathbf{F} \in \mathbb{C}^{m \times m}$ is the Fourier transform basis. If $|\alpha| + |\beta| < \sqrt{m}$, then (5.11) has a unique solution. Moreover, if $|\alpha| + |\beta| < \frac{1}{2}\sqrt{m}$, then the solution of (5.12) coincides with the unique solution of (5.11).

Donoho and Hue also discuss the utilization of Mutual Coherence (MC), see Definition 5.5, for deriving an uncertainty principle for real-valued signals and real-valued sinusoids in $\mathbf{F}$. These bounds were improved in [296] and a generalized uncertainty principle for arbitrary orthonormal bases was presented. The results are summarized in the following theorems.

Theorem 5.2 ([296]). Let $\mathbf{x} \in \mathbb{R}^m$ be a signal. Moreover, let $\Psi^{(1)}$ and $\Psi^{(2)}$ be orthonormal bases. The representation of the signal in $\Psi^{(1)}$ and $\Psi^{(2)}$, i.e. $\mathbf{x} = \Psi^{(1)} \mathbf{s}^{(1)} = \Psi^{(2)} \mathbf{s}^{(2)}$, obeys the inequality

$$\|\mathbf{s}^{(1)}\|_0 + \|\mathbf{s}^{(2)}\|_0 \geq \frac{2}{\mu([\Psi^{(1)}, \Psi^{(2)}])} \quad (5.14)$$

Theorem 5.3 ([296]). Assume that the signal $\mathbf{x} \in \mathbb{R}^m$ has a unique sparse representation $\mathbf{x} = [\Psi^{(1)}, \Psi^{(2)}] \mathbf{s}$. Then the solution of

$$\min_{\hat{\mathbf{s}}} \|\hat{\mathbf{s}}\|_0 \quad \text{s.t.} \quad \mathbf{x} = [\Psi^{(1)}, \Psi^{(2)}] \hat{\mathbf{s}}, \quad (5.15)$$

uniquely identifies this sparse representation if $\|\hat{\mathbf{s}}\|_0 < 1/\mu([\Psi^{(1)}, \Psi^{(2)}])$. Moreover, if the more stringent condition $\|\hat{\mathbf{s}}\|_0 < 0.9142/\mu([\Psi^{(1)}, \Psi^{(2)}])$ is satisfied, then the solution of ℓ_1 variant of (5.15) coincides with the solution of (5.15).

An illustration of bounds presented in Theorem 5.3 is given in Fig. 5.2. A formulation of the uncertainty principle for block-sparse signals can be found in [157].

5.5.2 Spark

In Section 5.5.1 we discussed the uniqueness conditions for measurement matrices of the form $[\Psi^{(1)}, \Psi^{(2)}]$, where $\Psi^{(1)}$ and $\Psi^{(2)}$ are orthonormal. However, as discussed in Section 1.2, measurement matrices are often defined by an overcomplete matrix $\mathbf{A} = \Phi \mathbf{D}$. Therefore, the matrix $\mathbf{A}$ is not necessarily a concatenation of orthonormal matrices; for instance let $\Phi \in \mathbb{R}^{\frac{m}{2} \times m}$ be any sub-sampling of rows from the identity matrix and $\mathbf{D}$ be any orthonormal matrix. The first tool to analyze such measurement matrices in the context of uniqueness was Spark [73], defined below.

Definition 5.8 (spark). *Spark of a matrix $\mathbf{A} \in \mathbb{R}^{s \times k}$, denoted $\text{spark}(\mathbf{A})$, is the smallest subset of columns in $\mathbf{A}$ that are linearly dependent.*

Spark of any matrix is bounded by $[2, \text{rank}(\mathbf{A}) + 1]$. The lower bound comes from the assumption that $\mathbf{A}$ does not have a zero-valued column. The upper bound is achieved for e.g. an orthonormal matrix. Note that a full-rank matrix may have a spark of 2. For example, $\text{spark}([\mathbf{1}, \mathbf{I}, \dots]) = 2$, where $\mathbf{1} \in \mathbb{R}^s$ is a vector of ones and $s \geq 2$. The matrix $[\mathbf{I}, \mathbf{F}]$ described in Section 5.5.1 can be shown to have a spark of $2\sqrt{m}$ [73]. Let $\Phi \in \mathbb{R}^{s \times m}$ be an overcomplete matrix with i.i.d. Gaussian elements and $\mathbf{D} \in \mathbb{R}^{m \times m}$ be an orthonormal matrix. Then $\text{spark}(\Phi \mathbf{D}) = s + 1$ [297]. While the spark of certain classes of matrices can be calculated analytically, computing the spark for general matrices is a combinatorial problem requiring 2^k steps for a matrix of size $s \times k$.

Spark can be used for determining the uniqueness of the solution obtained from an ℓ_0 problem, as well as the equivalence conditions between ℓ_0 and ℓ_1 solutions.

Theorem 5.4 ([73]). *Assume that a signal $\mathbf{y} \in \mathbb{R}^s$ has a unique sparse representation $\mathbf{y} = \mathbf{A}\mathbf{s}$, for a predefined equivalent measurement matrix $\mathbf{A} \in \mathbb{R}^{s \times k}$. Then the solution of*

$$\min_{\hat{\mathbf{s}}} \|\hat{\mathbf{s}}\|_0 \quad \text{s.t.} \quad \mathbf{y} = \mathbf{A}\hat{\mathbf{s}}, \quad (5.16)$$

is the unique sparsest solution if $\|\hat{\mathbf{s}}\|_0 < \text{spark}(\mathbf{A})/2$. Moreover, given this condition, the solution of the ℓ_1 variant of (5.16) coincides with the unique solution of (5.16).

The sparsity upper bound, $\|\hat{\mathbf{s}}\|_0 < \text{spark}(\mathbf{A})/2$, is sharp [73]; i.e. there is no sparsity bound $\delta > \text{spark}(\mathbf{A})/2$ that guarantees uniqueness. Moreover, spark and mutual coherence are related by the inequality $\text{spark}(\mathbf{A}) > 1/\mu(\mathbf{A})$.

5.5.3 Exact Recovery Coefficient

Exact Recovery Coefficient (ERC) was first introduced by J. Tropp in [79] as a metric for evaluating the performance of OMP and basis pursuit for solving ℓ_0

and ℓ_1 problems, respectively. Tropp formulates the recovery conditions for both τ -sparse and compressible signals. Later, these results were extended in [74] for noisy signals, and universal recovery conditions were derived. ERC is defined as follows

Definition 5.9 (ERC). *ERC of a matrix $\mathbf{A} \in \mathbb{R}^{s \times k}$ is given by*

$$\text{ERC}(\Lambda) = 1 - \max_{\alpha \in \Gamma} \|\mathbf{A}_\Lambda^\dagger \mathbf{A}_\alpha\|_1 \quad (5.17)$$

where $\Lambda \subset \{1, \dots, k\}$ and $\Gamma = \{1, \dots, k\} \setminus \Lambda$. The symbol $\setminus$ denotes set difference.

Consider the case where $\Lambda = \text{supp}(\mathbf{s})$ for the system $\mathbf{x} = \mathbf{A}\mathbf{s}$. Hence, ERC measures how distinct the atoms corresponding to Λ are compared to those that are not in the support. Indeed if there is a distinction, i.e. $\text{ERC}(\Lambda) > 0$, and the signal is noise free, then various recovery algorithms can find the support [79]. For the case of noisy signals, i.e. when the measurements are $\mathbf{y} = \mathbf{A}\mathbf{s} + \mathbf{w}$, where $\mathbf{w}$ is an unknown noise vector, Tropp [74] formulates the following theorem:

Theorem 5.5 ([74]). *Assume that an equivalent measurement matrix $\mathbf{A} \in \mathbb{R}^{s \times k}$ and a measured signal $\mathbf{y} \in \mathbb{R}^s$ are given. Let $\Lambda \subset \{1, \dots, k\}$ be an index set such that the columns of $\mathbf{A}_\Lambda$ are linearly independent and $\text{ERC}(\Lambda) > 0$. Then the solution $\hat{\mathbf{s}}$ of*

$$\min_{\hat{\mathbf{s}}} \|\hat{\mathbf{s}}\|_1 \quad \text{s.t.} \quad \|\mathbf{y} - \mathbf{A}\hat{\mathbf{s}}\|_2 \leq \epsilon, \quad (5.18)$$

is unique and $\text{supp}(\hat{\mathbf{s}}) \subset \Lambda$, if

$$\epsilon \geq \|\mathbf{y} - \mathbf{A}_\Lambda \mathbf{A}_\Lambda^\dagger \mathbf{y}\|_2 \sqrt{1 + \left(\frac{\delta \|\mathbf{A}_\Lambda^\dagger\|_{2,1}}{\text{ERC}(\Lambda)} \right)^2}, \quad (5.19)$$

where

$$\delta = \frac{\max_{i \in \{1, \dots, k\}} (\mathbf{y} - \mathbf{A}_\Lambda \mathbf{A}_\Lambda^\dagger \mathbf{y})^T \mathbf{A}_i}{\|\mathbf{y} - \mathbf{A}_\Lambda \mathbf{A}_\Lambda^\dagger \mathbf{y}\|}, \quad (5.20)$$

and $\|\cdot\|_{2,1}$ is defined in (5.3). Moreover, the representation error is bounded by

$$\|\hat{\mathbf{s}} - \mathbf{s}^*\|_2 \leq \delta \|\mathbf{A}_\Lambda^\dagger\|_F, \quad (5.21)$$

where $\mathbf{s}^*$ is the optimal sparse vector.

5.5.4 Restricted Isometry Constant

Restricted isometry constant was first introduced in [76] and it is defined as follows

Definition 5.10 (RIC). Let $\mathbf{A} \in \mathbb{R}^{s \times k}$ and $\tau \in \{1, \dots, s\}$. Restricted isometry constant δ_τ of $\mathbf{A}$ is the smallest number such that

$$(1 - \delta_\tau) \|\mathbf{s}\|_2^2 \leq \|\mathbf{A}\mathbf{s}\|_2^2 \leq (1 + \delta_\tau) \|\mathbf{s}\|_2^2, \quad (5.22)$$

holds for all τ -sparse signals $\mathbf{s}$.

Equation 5.22 can also be expressed as

$$\delta_\tau = \max_{\Lambda \subset \{1, \dots, s\}, |\Lambda| \leq \tau} \|\mathbf{A}_\Lambda^T \mathbf{A}_\Lambda - \mathbf{I}\|_{2,2}, \quad (5.23)$$

where $\|\cdot\|_{2,2}$ is the spectral norm of a matrix; i.e. $\|\mathbf{X}\|_{2,2} = \lambda_{\max}(\mathbf{X})$. RIC has several implications. From (5.23) we see that the smaller the RIC, δ_τ , the closer $\mathbf{A}_\Lambda$ is to an orthogonal matrix (for all possible sets $\Lambda \subset \{1, \dots, s\}$, $|\Lambda| \leq \tau$). Moreover, equation (5.23) shows that $\delta_2 = \mu(\mathbf{A})$ and that the singular values of $\mathbf{A}_\Lambda$ are in $[1 - \delta_\tau, 1 + \delta_\tau]$. Finally, according to (5.22), if $\delta_{2\tau}$ is sufficiently small, this means that the distances between τ -sparse signals under the transformation $\mathbf{A}$ is well preserved:

$$(1 - \delta_{2\tau}) \|\mathbf{s}^{(1)} - \mathbf{s}^{(2)}\|_2^2 \leq \|\mathbf{A}\mathbf{s}^{(1)} - \mathbf{A}\mathbf{s}^{(2)}\|_2^2 \leq (1 + \delta_{2\tau}) \|\mathbf{s}^{(1)} - \mathbf{s}^{(2)}\|_2^2. \quad (5.24)$$

Loosely speaking, we say that $\mathbf{A}$ satisfies the Restricted Isometry Property (RIP) if δ_τ is sufficiently small. When RIP holds, the solution of the ℓ_0 program is unique and coincides with the solution of the ℓ_1 program. More concretely,

Theorem 5.6 ([298]). Consider the measurement model $\mathbf{y} = \mathbf{A}\mathbf{s} + \mathbf{w}$, where $\mathbf{A} \in \mathbb{R}^{s \times k}$ and $\mathbf{w}$ is an unknown noise vector. Assume that $\delta_{2\tau} < \sqrt{2} - 1$ and $\|\mathbf{w}\|_2 \leq \epsilon$. Then the solution of

$$\min_{\hat{\mathbf{s}}} \|\hat{\mathbf{s}}\|_1 \quad \text{s.t.} \quad \|\mathbf{y} - \mathbf{A}\hat{\mathbf{s}}\|_2 \leq \epsilon, \quad (5.25)$$

satisfies

$$\|\mathbf{s} - \hat{\mathbf{s}}\|_2 \leq \frac{C_0}{\sqrt{\tau}} \|\mathbf{s} - \mathbf{s}_o\|_1 + C_1 \epsilon, \quad (5.26)$$

where C_0 and C_1 are constants that only depend on $\delta_{2\tau}$, and $\mathbf{s}_o$ is a solution obtained from the oracle estimator, i.e. one that knows the location of nonzero entries in $\mathbf{s}$.

The bound $\delta_{2\tau} < \sqrt{2} - 1$ has been improved several times [299, 300, 301, 302]. Although RIC is NP-hard to compute, it can be shown that certain classes of measurement matrices satisfy RIP, as described in Theorem 5.6. For instance, if the elements of $\mathbf{A}$ are i.i.d. Gaussian random variables with mean zero and variance $1/s$, then $\mathbf{A}$ satisfies RIP with overwhelming probability if $s \geq C\tau \log(k/\tau)$, where C is a constant [78]. The same applies if the columns of $\mathbf{A}$ are sampled uniformly at random on the unit sphere of $\mathbb{R}^s$.

Input: $\mathbf{y}$, $\mathbf{A}$, and a stopping criteria based on sparsity, τ , or an error threshold, ϵ .

Init.: Initialize the residual vector, $\mathbf{r}^0 = \mathbf{y}$, the set support set, $\Lambda = \emptyset$, the iteration number, $i = 1$.

Result: $\hat{\mathbf{s}}$

```

1 while  $i \leq \tau$  or  $\|\mathbf{r}^i\|_2 > \epsilon$  do
2    $\hat{\mathbf{s}} \leftarrow \mathbf{0}$ ;
3    $\lambda \leftarrow \max_j |\langle \mathbf{r}^{i-1}, \mathbf{A}_j \rangle|$ ;
4    $\Lambda^i \leftarrow \Lambda^{i-1} \cup \{\lambda\}$ ;
5    $\hat{\mathbf{s}}_{\Lambda^i} \leftarrow \mathbf{A}_{\Lambda^i}^+ \mathbf{y}$ ;
6    $\mathbf{r}^i \leftarrow \mathbf{y} - \mathbf{A} \hat{\mathbf{s}}$ ;
7    $i \leftarrow i + 1$ ;
8 end

```

Algorithm 5: Orthogonal Matching Pursuit (OMP)

5.6 Orthogonal Matching Pursuit (OMP)

Consider the measurement model $\mathbf{y} = \Phi \mathbf{x} + \mathbf{w}$ for a signal $\mathbf{x} \in \mathbb{R}^m$, where $\Phi \in \mathbb{R}^{s \times m}$ is a measurement matrix. Assume that $\mathbf{x} = \mathbf{D}\mathbf{s}$, where $\|\mathbf{s}\|_0 = \tau$ and $\mathbf{D} \in \mathbb{R}^{m \times k}$ is an overcomplete or orthonormal dictionary. The vector $\mathbf{w}$ represents unknown additive measurement noise. Moreover, let $\Lambda = \text{supp}(\mathbf{s})$. Orthogonal matching pursuit [126] is a greedy method, among many others [125, 127, 128, 129, 133, 134, 303], for solving the problem

$$\min_{\hat{\mathbf{s}}} \|\hat{\mathbf{s}}\|_0 \quad \text{s.t.} \quad \|\mathbf{y} - \mathbf{A}\hat{\mathbf{s}}\|_2 \leq \epsilon, \quad (5.27)$$

given the measurements $\mathbf{y}$ and the equivalent measurement matrix $\mathbf{A} = \Phi \mathbf{D}$. The constant ϵ is related to the noise vector $\mathbf{w}$. In many practical scenarios, the noise vector $\mathbf{w}$ is unknown. However, an estimate of its properties is typically available. Moreover, it is a common practice to assume the elements of $\mathbf{w}$ to be i.i.d. zero-mean Gaussian random variables with variance σ^2 , i.e. $\mathbf{w} \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$. As described in Section 1.1, greedy methods estimate Λ iteratively. In fact, the main difference between various greedy algorithms lies in their approach to the support estimation problem. The OMP algorithm, in particular, estimates one support index at each iteration.

Algorithm 5 summarizes the steps taken by OMP for identifying $\hat{\mathbf{s}}$ and $\text{supp}(\hat{\mathbf{s}})$. Here, Λ^i defines the support of $\hat{\mathbf{s}}$ at iteration i , i.e. $\Lambda^i = \text{supp}(\hat{\mathbf{s}}^i)$. Since one coefficient in $\hat{\mathbf{s}}$ is recovered at each iteration, we have that $|\Lambda^i| = i$. The problem of estimating the support of $\mathbf{s}$ is equivalent to finding k columns in $\mathbf{A}$ that best

Figure 5.3: A geometrical illustration for the first iteration of OMP given that $\mathbf{A}$ has four atoms.

approximate the input signal $\mathbf{y}$. In addition, the matrix $\mathbf{A}_{\Lambda^i}$ defines a linear subspace spanned by the columns of $\mathbf{A}$ that are indexed by Λ^i . At each iteration, the dimensionality of this subspace increases until the error of representing the signal in the subspace is smaller than ϵ , or the dimensionality of the subspace is smaller than τ . Therefore, the iterations may be terminated according to two criteria. If the sparsity, τ , is known or can be estimated, the algorithm is terminated once we have τ elements in $\hat{\mathbf{s}}$. Alternatively, the iterations are terminated once the residual becomes sufficiently small. In practice, the former criteria is used more often since as Λ^i grows, the computational complexity of the algorithm grows (see step 5 of the algorithm).

We now analyze each step of the algorithm and provide a geometrical interpretation, which sheds light onto the intuition behind each step. Starting at iteration zero, we have $\mathbf{r}^0 = \mathbf{y}$. The inner product between the measurement vector $\mathbf{y}$ and all the atoms in $\mathbf{A}$ is calculated at line 3 of Algorithm 5. The index of the atom with largest inner product value is added to the support set, Λ . Indeed this operation chooses an atom with the highest correlation to the measurement vector. This is shown in Fig. 5.3a, where we assume that $\mathbf{A}$ has 4 atoms. In this example, atom number 3 has the highest correlation with $\mathbf{y}$. Next, we calculate the current estimate of sparse coefficients by solving the least squares problem $\|\mathbf{y} - \mathbf{A}_{\Lambda^i} \hat{\mathbf{s}}_{\Lambda^i}\|_2^2$. The solution is stated at line 5 of Algorithm 5. In geometrical terms, the measurement vector $\mathbf{y}$ is projected onto the column space of $\mathbf{A}_{\Lambda}$ to obtain the representation $\hat{\mathbf{s}}_{\Lambda^i}$; this is illustrated in Fig. 5.3b. Finally, we update the residual vector at line 6. It can be shown that [80]

$$\mathbf{r}^i = \mathbf{y} - \mathbf{A} \hat{\mathbf{s}} = \mathbf{y} - \mathbf{A}_{\Lambda^i} \mathbf{A}_{\Lambda^i}^\dagger \mathbf{y} = (\mathbf{I} - \mathbf{P}_{\Lambda^i}) \mathbf{y} = \mathbf{P}_{\Lambda^i}^\perp \mathbf{y}, \quad (5.28)$$

where $\mathbf{P}_{\Lambda^i}$ is the orthogonal projection operator onto the column space of $\mathbf{A}_{\Lambda^i}$; moreover, $\mathbf{P}_{\Lambda^i}^\perp$ is the orthogonal projection operator onto the orthogonal complement of the column space of $\mathbf{A}_{\Lambda^i}$. This is illustrated in Fig. 5.3c, where the residual

Figure 5.4: The effect of noise vector $\mathbf{w}$ on step 3 of algorithm 5 for the first iteration is shown. The noise vector $\mathbf{w}$ causes OMP to wrongly select atom 2, while the correct atom is 3.

vector is orthogonal to atom number 3. Given the new residual vector, we go back to step 1 and repeat the above mentioned procedure. It is clear from Fig. 5.3c that at the next iteration, atom number 1 is selected.

The OMP algorithm has been shown to be very effective in practice for sparse signals [66, 304, 305]. Even though there exists various extensions of the algorithm, OMP provides a good trade off between the quality and the speed of reconstruction. Moreover, OMP is simple to implement, as it is evident from Algorithm 5. The simplicity of OMP has enabled hardware implementations on Field-Programmable Gate Array (FPGA) circuits [306, 307, 308], as well as GPU implementations [243, 308]. However, the performance of OMP in terms of speed and accuracy degrades for signals that are not sufficiently sparse. For instance, compressible signals are well approximated using ℓ_1 methods rather than greedy algorithms such as OMP.

5.7 Theoretical Analysis of OMP Using Mutual Coherence

In this section, we present a novel analysis on the performance of OMP in correctly identifying the support of a sparse signal. The materials presented herein are a combination of the results reported in Paper F and Paper G. We start by discussing the effect of measurement noise on the performance of OMP in Section 5.7.1. Next, we present related work on the topic in Section 5.7.2, where we also describe

why mutual coherence is a suitable candidate for the analysis of OMP. Our novel analysis of OMP is presented in Section 5.7.3, followed by numerical results in Section 5.7.4.

5.7.1 The Effect of Noise on OMP

An important aspect affecting the accuracy of OMP is the measurement noise, $\mathbf{w}$. A large amount of noise can lead to incorrect atom selection (at line 3 of Algorithm 5). Figure 5.4b demonstrates this effect for iteration 0. The Gaussian noise is shown with a green circle, where the CDF is illustrated with a color gradient. Because the noise is additive, i.e. $\mathbf{y} = \mathbf{A}\mathbf{s} + \mathbf{w}$, it moves the residual vector to a new random point (shown in blue). As it can be seen, at this new location, OMP will choose atom 2 instead of 3. Hence, it is essential to quantify the tolerance of OMP to noise for correct atom selection. This is one of the contributions of this thesis, which will be described in detail in Section 5.7.3. Figure 5.4a shows the empirical probability of support recovery for OMP with respect to the standard deviation of the noise vector. In this plot, for a fixed value of σ , the probability of success is calculated as the number of successful support recoveries among 5000 trials, where in each trial we instantiate a new sparse signal and noise vector. It can be seen that increasing the noise quickly decreases the probability of support recovery.

Since greedy algorithms solve an ℓ_0 problem, e.g. equation (5.27), it cannot be guaranteed (in general) that an optimal solution can be obtained. This is in contrast with ℓ_1 algorithms, which solve a convex problem with a guaranteed convergence to the global minimum. Moreover, the addition of noise to measurements further complicates the analysis of convergence. As described in Section 5.6, OMP is computationally efficient and competitive in terms of accuracy, while enabling simple implementations. As a result, a substantial amount of work has been dedicated for obtaining theoretical guarantees for the convergence of OMP.

5.7.2 Prior Art and Motivation

Tropp analyzed the convergence of OMP using ERC in [79]. Various extensions of this work exists. For instance, partial-ERC was introduced in [309, 310] for solving (5.27) when the support is partially known. Restricted isometry property has been used extensively for the analysis of OMP [80, 311, 312, 313, 314]. Wang and Shim [315] utilize RIP to derive the required number of OMP iterations for exact recovery and Ding et al. [316] derive sufficient conditions using RIP when both the signal $\mathbf{y}$ and the measurement matrix $\mathbf{A}$ are perturbed (noisy). Spark has not directly been used for OMP analysis. However, performance guarantees using spark (e.g. Theorem 5.4) do apply to OMP.

While RIC, ERC, and spark provide accurate performance guarantees, and in some cases even sharp conditions, these constants cannot be computed in practice for a large family of measurement matrices. Computing ERC is a combinatorial problem that is often intractable for large measurement matrices. In general, computing RIC and spark for arbitrary measurement matrices is NP-hard [317]. We discussed in Section 5.5 that these constants can only be computed (or bounded) for a small subset of random or deterministic matrices. In contrast, mutual coherence is simple to compute and has a complexity of $\mathcal{O}(sk^2)$ for a matrix $\mathbf{A} \in \mathbb{R}^{s \times k}$. This is particularly important for training-based dictionaries, since it is difficult, and most of the times impossible, to compute RIC, ERC, and spark. For instance, assume that $\mathbf{D}$ is trained using the K-SVD algorithm (described in Section 2.4), and Φ is a spike ensemble. The equivalent sensing matrix, $\mathbf{A}$, is deterministic and there are no closed-form solutions for obtaining RIP, ERC, or spark.

One of the early works on deriving coherence-based performance guarantees for OMP was presented in [18]. Theorem 5.7 below presents the main result of [18]. Without loss of generality, we assume that the columns of $\mathbf{A}$ are normalized.

Theorem 5.7 ([18]). *Let $\mathbf{y} = \mathbf{A}\mathbf{s} + \mathbf{w}$, where $\mathbf{A} \in \mathbb{R}^{s \times k}$, $\|\mathbf{s}\|_0 = \tau$ and $\mathbf{w} \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$. Define $\mathbf{s}_{\min} = \min_j |\mathbf{s}_j|$. If*

$$\mathbf{s}_{\min} - (2\tau - 1)\mu(\mathbf{A})\mathbf{s}_{\min} \geq 2\beta, \quad (5.29)$$

where $\beta \triangleq \sigma\sqrt{2(1+\alpha)\log k}$ is defined for some constant $\alpha > 0$, then with probability at least

$$1 - \frac{1}{k^\alpha \sqrt{\pi(1+\alpha)\log k}}, \quad (5.30)$$

OMP identifies the true support of $\mathbf{s}$.

The proof of Theorem 5.7 is based on analyzing the event $\Pr\{|\langle \mathbf{A}_j, \mathbf{w} \rangle| \leq \beta\}$, $\forall j \in \{1, \dots, k\}$. Hence the correlation of the noise vector with the atoms of $\mathbf{A}$ is considered. Indeed if this correlation is too large, i.e. when the noise vector dominates the signal, then step 3 in Algorithm 5 will select an incorrect atom (see Fig. 5.4b). Hence, Ben-Haim et al. [18] analyze the probability that the noise correlation is bounded. Equation (5.30) is in fact a lower bound for the event $\Pr\{|\langle \mathbf{A}_j, \mathbf{w} \rangle| \leq \beta\}$. Therefore, we see that the probability of support recovery, (5.30), is decoupled from important parameters related to $\mathbf{s}$ and $\mathbf{A}$ such as $\mathbf{s}_{\min}$, τ , and $\mu(\mathbf{A})$. These parameters affect the condition in (5.29); hence if the condition fails, equation (5.30) is invalid. Moreover, $\mathbf{s}_{\max} = \max_j |\mathbf{s}_j|$ is not considered. In many applications, $\mathbf{s}_{\max}$ and $\mathbf{s}_{\min}$ are known or can be estimated. For instance, in imaging applications one can obtain bounds on the values of the sparse signal $\mathbf{s}$ because the original signal, $\mathbf{x}$, is typically in $[0, 1]$ to facilitate quantization. The

ratio of the largest and smallest signal values is known as the *Dynamic Range* (DR) of a signal.

5.7.3 A Novel Analysis Based on the Concentration of Measure

Paper F presents a performance guarantee for the exact support recovery of a noisy signal using OMP and mutual coherence. By identifying the shortcomings in the results of [18], the proposed analysis contains signal parameters in the probability bound and is shown to better match the empirical results. We state the main result below.

Theorem 5.8 (Paper F). *Let $\mathbf{y} = \mathbf{A}\mathbf{s} + \mathbf{w}$, where $\mathbf{A} \in \mathbb{R}^{s \times k}$, $\tau = \|\mathbf{s}\|_0$ and $\mathbf{w} \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$. Moreover, assume that the nonzero elements of $\mathbf{s}$ are independent centered random variables with an arbitrary distribution. If $\mathbf{s}_{\min}/2 \geq \beta$, for some constant $\beta \geq 0$, then OMP identifies the true support with lower bound probability*

$$\left(1 - k \sqrt{\frac{2}{\pi}} \frac{\sigma}{\beta} e^{-\beta^2/2\sigma^2}\right) \left(1 - 2k \exp\left(\frac{-k(\mathbf{s}_{\min}/2 - \beta)^2}{2\tau^2\gamma^2 + 2k\gamma(\mathbf{s}_{\min}/2 - \beta)/3}\right)\right), \quad (5.31)$$

where $\gamma = \mu(\mathbf{A})\mathbf{s}_{\max}$.

Before presenting the proof, we will summarize important implications of Theorem 5.8 in the following remarks.

Remark 5.1. *A key factor that distinguishes Theorem 5.8 from Theorem 5.7 is the assumption of i.i.d. random variables in $\mathbf{s}$, as opposed to the assumption of deterministic variables. This is by no means a limiting factor since we do not assume the distribution of these variables. Most importantly, randomness of nonzero values in $\mathbf{s}$ enables us to use concentration inequalities in the proof of Theorem 5.8, which greatly simplifies the analysis and provides a more accurate probability bound.*

Remark 5.2. *The probability bound in (5.31) contains two terms: The first term is related to the noise properties and the signal length, while the second term is related to the signal and dictionary properties. This is in sharp contrast with Theorem 5.7, where a condition is imposed on the signal and dictionary properties, while the probability bound only considers the noise related factors and the signal length.*

Remark 5.3. *The condition $\mathbf{s}_{\min}/2 \geq \beta$ of Theorem (5.8) is far less stringent than (5.29). Moreover, numerical simulations show that $\mathbf{s}_{\min}/2 \geq \beta$ is satisfied for any reasonable choice of parameters.*

Remark 5.4. *We observe that the first term of (5.31) is equivalent to (5.30). This can be shown by replacing β in the first term of (5.31) with $\sigma\sqrt{2(1+\alpha)\log k}$, which*

is the value used for β in Theorem 5.7. Since the second term of (5.31) is in $[0, 1]$, this implies that (5.30) is always larger than (5.31). However, the condition imposed by Theorem 5.7 is not satisfied for a large range of parameters. This means that (5.30) is not valid in many cases, for which (5.31) may evaluate to high probabilities.

A sketch of the proof for Theorem 5.8 is presented below and the reader is referred to Paper F for more details.

Proof sketch for Theorem 5.8. Ben-Haim et al. show that if the event $\Pr\{|\langle \mathbf{A}_j, \mathbf{w} \rangle| \leq \beta\}$ holds, then OMP identifies the true support, Λ , if [18]

$$\min_{j \in \Lambda} |\langle \mathbf{A}_j, \mathbf{A}_\Lambda \mathbf{s}_\Lambda + \mathbf{w} \rangle| \geq \max_{k \notin \Lambda} |\langle \mathbf{A}_k, \mathbf{A}_\Lambda \mathbf{s}_\Lambda + \mathbf{w} \rangle|. \quad (5.32)$$

Simple manipulations of (5.32) yields

$$\begin{cases} \max_{i \notin \Lambda} \{\Gamma_i\} < \min_{j \in \Lambda} \frac{|\mathbf{s}_j|}{2}, \\ \max_{j \in \Lambda} |\langle \mathbf{A}_j, \mathbf{A}_{\Lambda \setminus \{j\}} \mathbf{s}_{\Lambda \setminus \{j\}} + \mathbf{w} \rangle| < \min_{j \in \Lambda} \frac{|\mathbf{s}_j|}{2}, \end{cases} \quad (5.33)$$

where $\Gamma_i = |\langle \mathbf{A}_i, \mathbf{A}_\Lambda \mathbf{s}_\Lambda + \mathbf{w} \rangle|$. In fact we have broken (5.32) into two (joint) conditions for successful recovery. The first one checks whether the maximum correlation of atoms in the support with $\mathbf{y}$ (or the residual) is smaller than $\mathbf{s}_{\min}$. The second one checks whether the maximum correlation of atom j in the support with the measurement vector $\mathbf{y}$, *excluding* atom j , is smaller than $\mathbf{s}_{\min}$. From (5.33), an upper bound for the probability of error can be obtained

$$\Pr\{\text{error}\} \leq \sum_{j \in \Lambda} \Pr\left\{|\langle \mathbf{A}_j, \mathbf{A}_{\Lambda \setminus \{j\}} \mathbf{s}_{\Lambda \setminus \{j\}} + \mathbf{w} \rangle| \geq \frac{\mathbf{s}_{\min}}{2}\right\} + \sum_{i \notin \Lambda} \Pr\left\{\Gamma_i \geq \frac{\mathbf{s}_{\min}}{2}\right\}. \quad (5.34)$$

We now bound the two terms on the right-hand side of (5.34), excluding the summations. For the second term we have

$$\Pr_{i \notin \Lambda} \left\{ \Gamma_i \geq \frac{\mathbf{s}_{\min}}{2} \right\} \leq 2 \exp \underbrace{\left(\frac{-(\mathbf{s}_{\min}/2 - \beta)^2}{2(\frac{\tau^2 \gamma^2}{k} + \frac{\gamma(\mathbf{s}_{\min}/2 - \beta)}{3})} \right)}_{P_2}. \quad (5.35)$$

Equation (5.35) is obtained by expanding Γ_i to form a sum of random variables, followed by the application of the Bernstein inequality [318], which is one of the several concentration inequalities [319] for sum of random variables. A similar

Figure 5.5: Numerical simulations for comparing theorems 5.7 and 5.8 with empirical support recovery performance of OMP. For the left column of plots we set $s = 1024$, and for the right column $s = 2048$. Moreover, $k = 2s$ and $s_{\max} = 1.0$.

bound can be found for the first term on the right-hand side of (5.34), which we denote by P_1 . Given P_1 and P_2 , we can rewrite (5.34) as

$$\Pr\{\text{error}\} \leq \tau P_1 + (k - \tau) P_2 \leq k P_2, \quad (5.36)$$

because $P_2 > P_1$. Inverting $\Pr\{\text{error}\}$ gives the second term of (5.31), which completes the proof. $\blacksquare$

5.7.4 Numerical Results

Numerical simulations for comparing Theorem 5.8 and Theorem 5.7, as well as empirical results of OMP are summarized in Fig. 5.5. The empirical probability was obtained by performing OMP 5000 times, where in each trial a random sparse signal (i.e. random support and random nonzero elements) and a random noise vector $\mathbf{w}$ were constructed. The probability of success is the ratio of successful trials in identifying the support of $\mathbf{s}$ to the total number of trials. The measurement matrix used here is $\mathbf{A} = [\mathbf{I}, \mathbf{H}]$, where $\mathbf{H}$ is the Hadamard matrix. For theorems 5.7 and 5.8, we fix the value of β to facilitate comparison. It can be seen that the condition (5.29) is not satisfied for a large range of parameters, specifically when $s_{\min}$ is small (i.e. high dynamic range), τ is large, or σ is large. Most importantly, the shape of the probability curve produced by Theorem 5.8 is similar to that of OMP, while Theorem 5.7 produces a step function.

5.7.5 User-friendly Bounds

Theorem 5.8 states the probability of recovering the support of $\mathbf{s}$. In absence of noise, once we have recovered the correct support, the signal $\mathbf{s}$ can be exactly reconstructed (see (2.14)). However, in presence of noise, the estimates of nonzero signal values, s_Λ , are not exact. Fortunately, we can derive an estimate of the error introduced by the noise. The following theorem states an upper bound for the absolute error of the solution obtained from OMP, assuming that the support is recovered correctly.

Theorem 5.9 ([65] and Paper G). *Using the same setup as in Theorem 5.8, assume that OMP identifies the true support of $\mathbf{s}$. Then the error of the estimated sparse signal with respect to $\mathbf{s}$ is upper bounded by*

$$E_{abs} \triangleq \|\mathbf{s} - \hat{\mathbf{s}}\|_2^2 \leq \frac{\tau\beta^2}{(1 - \mu_{\max}(\tau - 1))^2}, \quad (5.37)$$

In what follows, it will be shown that by combining theorems 5.8 and 5.9, we can obtain approximate, but *user-friendly*, bounds that depend on the parameters commonly used in practice. Moreover, this analysis sheds light on the effect of different parameters on the two terms of (5.31), see Remark 5.2. The results reported in this section are from Paper G; however, here we reformulate these results for real-values signals.

Define the dynamic range, Signal to Noise Ratio (SNR), and the relative error, respectively as follows

$$\text{DR} \triangleq \left(\frac{s_{\max}}{s_{\min}} \right)^2, \quad (5.38)$$

$$\text{SNR} \triangleq \left(\frac{s_{\min}}{\sigma} \right)^2, \quad (5.39)$$

$$E_{\text{rel}} \triangleq \frac{E_{\text{abs}}}{\sum_{n=1}^N \mathbf{s}_n} \leq \frac{E_{\text{abs}}}{\tau s_{\min}^2}. \quad (5.40)$$

Using simple calculations, Paper G shows that (5.31) can be approximated as

$$\left(1 - k \sqrt{\frac{2}{\pi \rho^2 E_{\text{rel}} \text{SNR}}} e^{-E_{\text{rel}} \rho^2 \text{SNR}/2} \right) \left(1 - 2k \exp \left(\frac{-1}{\frac{8\tau^2 \mu^2 \text{DR}}{k} + \frac{4\mu \sqrt{\text{DR}}}{3}} \right) \right), \quad (5.41)$$

where we define $\rho^2 = (1 - \mu(\tau - 1))^2$ and $\mu = \mu(\mathbf{A})$ for notational brevity. Assume that SNR is very high, then the first term of (5.41) becomes close to 1 and can be omitted. In this case, the probability of support recovery is lower bounded by

$$1 - 2k \exp \left(\frac{-1}{\frac{8\tau^2 \mu^2 \text{DR}}{k} + \frac{4\mu \sqrt{\text{DR}}}{3}} \right). \quad (5.42)$$

We have now obtained a bound that is only dependent on the signal properties and the equivalent measurement matrix. Let us now consider the case when μ is very small. Note that $\mu = 0$ only holds for matrices with orthogonal columns. A lower bound for mutual coherence of overcomplete matrices is $\sqrt{(k-s)[s(k-1)]^{-1}}$. For instance, a two times overcomplete matrix $\mathbf{A} \in \mathbb{R}^{1000 \times 2000}$, has a mutual coherence of at least $\mu(\mathbf{A}) = 2.24\text{e-}5$. This lower bound can be achieved for e.g. $\mathbf{A} = [\mathbf{I}, \mathbf{F}]$ or $\mathbf{A} = [\mathbf{I}, \mathbf{H}]$, where $\mathbf{F}$ is the Fourier transform matrix and $\mathbf{H}$ is the Hadamard matrix. When mutual coherence is small, we can approximate (5.41) as

$$1 - N \sqrt{\frac{2}{\pi E_{\text{rel}} \text{SNR}}} e^{-E_{\text{rel}} \text{SNR}/2}. \quad (5.43)$$

As it can be seen, equation (5.43) is independent of DR. This is expected because when the mutual coherence is low, atoms of $\mathbf{A}$ are far apart, so that the DR does not affect the atom selection process.

5.8 Uniqueness Under a Dictionary Ensemble

So far we have described selected fundamental results in compressed sensing for *one* dictionary (Section 5.5), as well as some of the contributions of this

thesis regarding exact recovery with OMP for signals that are sparse under an overcomplete dictionary (Section 5.7). These theoretical results suggest that the higher the sparsity of a signal, the higher the probability of exact recovery. As described in Chapter 1, natural signals are not typically sparse, but compressible. In practical applications, small coefficients of a compressible signal are nullified to obtain a sparse signal, which indeed introduces error in the representation. For a fixed signal (e.g. an image), it is the effectiveness of the dictionary that defines the representation error. The ensemble of dictionaries introduced in sections 3.2 and 3.3 was empirically shown to promote sparsity by performing consecutive clustering and dictionary training. As it will be seen in Chapter 6, this increase in sparsity improves the performance of compressed sensing. It remains to be seen that when a dictionary ensemble is used for compressed sensing, can the uniqueness of the obtained solution be guaranteed? How many samples are needed for exact recovery? These questions are answered in Paper D and Paper B, which are also the focus of this section.

In Section 5.8.1, we formulate a compressed sensing problem for the case when the unknown signal that we would like to reconstruct is sparse in an ensemble of dictionaries. Next, in Section 5.8.2, we define uniqueness for the solution obtained from an ensemble of dictionaries using compressed sensing. A key theoretical contribution of this thesis regarding the aforementioned problem is presented in Section 5.8.3. Finally, we discuss the results of numerical simulations obtained from the proposed theoretical analysis in Section 5.8.4.

5.8.1 Problem Formulation

Let $\Psi = \{\mathbf{D}^{(d)}\}_{d=1}^D$ be an ensemble of dictionaries, where $\mathbf{D}^{(d)} \in \mathbb{R}^{m \times k}$. A training based method for obtaining an ensemble of multidimensional orthonormal dictionaries was discussed in Section 3.3. In this section, we do not assume any structure on these dictionaries; i.e. each dictionary can be orthogonal, overcomplete, with i.i.d. Gaussian entries, learned, deterministic, or analytical. Furthermore, we convert multidimensional dictionaries to their equivalent 1D form. This can be simply achieved using the Kronecker product as follows

$$\mathbf{D}^{(d)} \triangleq \mathbf{U}^{(n,d)} \otimes \dots \otimes \mathbf{U}^{(1,d)}, \forall d \in \{1, \dots, D\}, \quad (5.44)$$

where $\{\mathbf{U}^{(1,d)}, \dots, \mathbf{U}^{(n,d)}\}_{d=1}^D$ is an ensemble of nD dictionaries. Hence, the ℓ_1 sparse recovery problem using an ensemble can be formulated as

$$\min_{\hat{\mathbf{s}}, \mathbf{D} \in \Psi} \|\hat{\mathbf{s}}\|_1 \quad \text{s.t.} \quad \|\mathbf{x} - \mathbf{D}\hat{\mathbf{s}}\|_2 \leq \epsilon. \quad (5.45)$$

In other words, we need to solve (5.45) for all the dictionaries in Ψ and pick the most sparse solution. Indeed a successful recovery requires that the signal $\mathbf{x}$ has

a unique sparse representation among *all* the dictionaries. In other words, one dictionary in Ψ should lead to the most sparse representation. If two or more dictionaries produce a solution with the same sparsity for $\mathbf{x}$, then uniqueness cannot be guaranteed. Uniqueness of the solution of (5.45) will be defined more concretely in Section 5.8.2.

We can (theoretically) construct an ensemble such that a signal $\mathbf{x}$ has a unique sparse representation. The training algorithm described in Section 3.2 learns an ensemble such that a signal is sparse in only one dictionary. However, the optimization problem is highly nonlinear, hence being trapped in a local minima is inevitable. Note that (5.45) is a sparse representation problem rather than compressed sensing. Even if we can construct an optimal ensemble, the equivalent measurement matrix $\mathbf{A} = \Phi \mathbf{D}^{(d)}$ may lead to multiple sparse representations that hinders the uniqueness of the solution obtained from

$$\min_{\mathbf{s}, \mathbf{D} \in \Psi} \|\hat{\mathbf{s}}\|_1 \quad \text{s.t.} \quad \|\mathbf{y} - \Phi \mathbf{D} \hat{\mathbf{s}}\|_2 \leq \epsilon, \quad (5.46)$$

where $\mathbf{y} = \Phi \mathbf{x}$.

5.8.2 Uniqueness Definition

We now formulate the uniqueness conditions for a dictionary ensemble Ψ . We assume that there exists a unique solution to problem (5.46) for any of the dictionaries in Ψ . This can be verified with the metrics described in Section 5.5. For instance, using spark, it is required that $\text{spark}(\Phi \mathbf{D}^{(i)}) > 2\tau$ for any dictionary $\mathbf{D}^{(i)} \in \Psi$ and τ -sparse signal $\mathbf{x}$. Without this assumption, guaranteeing uniqueness for all the dictionaries can quickly become intractable. Let $\mathbf{D}^{(i)}$ and $\mathbf{D}^{(j)}$ be any two dictionaries in Ψ . Assume that $\mathbf{x}$ has a unique sparse representation in $\mathbf{D}^{(i)}$, i.e. $\mathbf{x} = \mathbf{D}^{(i)} \mathbf{s}^{(i)}$, where $\|\mathbf{s}\|_0 = \tau$ and $\mathbf{y} = \Phi \mathbf{D}^{(i)} \mathbf{s}^{(i)}$. Uniqueness of the solution of (5.46) is guaranteed if there is no $\bar{\mathbf{x}} \neq \mathbf{x}$, where $\bar{\mathbf{x}} = \mathbf{D}^{(j)} \mathbf{s}^{(j)}$ and $\|\mathbf{s}^{(j)}\|_0 = \tau$, that also satisfies $\mathbf{y} = \Phi \mathbf{D}^{(j)} \mathbf{s}^{(j)}$. Gleichman and Eldar [297] propose the following metric to verify the uniqueness under a dictionary ensemble.

Definition 5.11 (τ -rank preserving [297]). *A measurement matrix Φ is called τ -rank preserving of a dictionary ensemble Ψ if for any index sets I and J with cardinality not larger than τ , and any two dictionaries $\mathbf{D}^{(i)} \in \Psi$ and $\mathbf{D}^{(j)} \in \Psi$, the following equality holds*

$$\text{rank} \left(\Phi \begin{bmatrix} \mathbf{D}_I^{(i)} & \mathbf{D}_J^{(j)} \end{bmatrix} \right) = \text{rank} \left(\begin{bmatrix} \mathbf{D}_I^{(i)} & \mathbf{D}_J^{(j)} \end{bmatrix} \right). \quad (5.47)$$

Verifying Definition 5.11 is indeed difficult since all the possibilities of I and J for any two dictionaries have to be assessed. Gleichman and Eldar [297] show that an

i.i.d. Gaussian matrix Φ is almost surely τ -rank preserving of an ensemble of bases (matrices with linearly independent columns). Formally,

Theorem 5.10 ([297]). *Assume that $\Phi \in \mathbb{R}^{s \times m}$, where $s \leq m$, is a matrix with i.i.d. Gaussian entries and $\Psi = \left\{ \mathbf{D}^{(d)} \in \mathbb{R}^{m \times k} \right\}_{d=1}^D$ is a finite set of bases. Then, Φ is almost surely τ -rank preserving of the set Ψ . Furthermore, if $\text{spark}(\Phi \mathbf{D}) > 2\tau$ for any dictionary $\mathbf{D} \in \Psi$, and Φ is τ -rank preserving, then the most sparse solution of (5.46), denoted $\mathbf{s}^*$, gives the unique solution $\hat{\mathbf{x}} = \mathbf{D}^* \mathbf{s}^*$, where $\mathbf{D}^* \in \Psi$ is the dictionary that gives $\mathbf{s}^*$.*

It remains to be seen if other types of sensing matrices are τ -rank preserving.

5.8.3 Uniqueness with a Spike Ensemble

In Paper D, we present a new analysis for uniqueness under an ensemble of dictionaries when a spike ensemble (see Definition 5.7) is used for sensing. We first observe that a spike ensemble does not satisfy (5.47) for an arbitrary dictionary ensemble, and therefore uniqueness cannot be guaranteed. Hence, a probabilistic analysis for uniqueness is proposed, where a lower bound on successful recovery based on the number of samples and the ensemble Ψ is derived. Moreover, the new analysis removes the requirement of having an ensemble of bases. As a result, even an ensemble of overcomplete dictionaries can be used. Before presenting the main result below, a few notations are introduced. The function $\text{orth}(\mathbf{X})$ defines the orthogonalization of the columns of $\mathbf{X}$, using e.g. QR decomposition [217]. We use $\text{supp}(\mathbf{p})_{\mathbf{D}}$ to denote the support of a signal $\mathbf{p}$ under a dictionary $\mathbf{D}$. Indeed a fixed signal has variable support under different dictionaries.

Theorem 5.11 (Paper D). *Assume that a signal $\mathbf{x} \in \mathbb{R}^m$ is at most τ -sparse under all dictionaries in $\Psi = \left\{ \mathbf{D}^{(d)} \in \mathbb{R}^{m \times k} \right\}_{d=1}^D$, where*

$$\tau \leq \frac{1}{2} \left(1 + \frac{1}{\mu(\mathbf{I}_{\Lambda, \cdot}, \mathbf{D})} \right), \forall \mathbf{D} \in \Psi, \quad (5.48)$$

or

$$\tau < \frac{\text{spark}(\mathbf{I}_{\Lambda, \cdot}, \mathbf{D})}{2}, \forall \mathbf{D} \in \Psi, \quad (5.49)$$

is satisfied for a fixed spike ensemble $\mathbf{I}_{\Lambda, \cdot} \in \mathbb{R}^{s \times m}$. Let

$$\alpha = \max_{j=1, \dots, m} \left\| \text{orth}([\mathbf{A}_{\cdot, I}, \mathbf{B}_{\cdot, J}])_{j, \cdot} \right\|_2^2, \forall \mathbf{A}, \forall \mathbf{B} \in \Psi, \mathbf{A} \neq \mathbf{B}, \quad (5.50)$$

$$r = \text{rank}([\mathbf{A}_{\cdot, I}, \mathbf{B}_{\cdot, J}]), \quad (5.51)$$

$$I = \text{supp}(\mathbf{x})_{\mathbf{A}}, \mathbf{A} \in \Psi, \quad (5.52)$$

$$J = \text{supp}(\mathbf{x})_{\mathbf{B}}, \mathbf{B} \in \Psi. \quad (5.53)$$

If

$$s \geq -m\alpha \log\left(\frac{1}{r}\right), \quad (5.54)$$

then with probability at least

$$1 - r(0.3679)^{s/m\alpha}, \quad (5.55)$$

the sparsest solution of (5.46), denoted $\mathbf{s}^*$, gives the unique solution $\hat{\mathbf{x}} = \mathbf{D}^* \mathbf{s}^*$, where $\mathbf{D}^* \in \Psi$ is the dictionary corresponding to $\mathbf{s}^*$.

The condition on sparsity, i.e. (5.48) or (5.49), ensures that there exists a unique solution for each optimization problem in (5.46). Indeed other metrics such as RIP (Section 5.5.4) or ERC (Section 5.5.3) can be used for this purpose. Moreover, Theorem 5.11 presents a universal result. In other words, regardless of the algorithm used to solve each optimization problem in (5.46), or its ℓ_0 variant, the probability of uniqueness is given by (5.55), as long as the number of samples is adequate in the spike ensemble. A sketch of the proof is presented below and the reader is referred to Paper D or the supplementary materials of Paper B for more details.

proof sketch for Theorem 5.11. For notational brevity, define $\mathbf{G} = [\mathbf{A}_{\cdot, I}, \mathbf{B}_{\cdot, J}]$. Because $\text{rank}(\mathbf{G}) = r$, it is enough to show that $\mathbf{I}_{\Lambda, \cdot} \mathbf{G}$ is full rank (note that $s \geq 2\tau \geq r$), which by Theorem 5.10 proves uniqueness.

It can be shown that the Gram matrix of $\mathbf{I}_{\Lambda, \cdot} \mathbf{G}$ can be written as sum of positive semidefinite matrices:

$$\sum_{j \in \Lambda} \underbrace{\mathbf{G}_{j, \cdot}^T \mathbf{G}_{j, \cdot}}_{\mathbf{X}_j} \quad (5.56)$$

Because the index set Λ is constructed by uniform random sampling of the set $\{1, \dots, m\}$, we can assume that the Gram matrix of $\mathbf{I}_{\Lambda, \cdot} \mathbf{G}$ can be expressed by sum of random matrices sampled without replacement from the finite set $\{\mathbf{X}_j = \mathbf{G}_{j, \cdot}^T \mathbf{G}_{j, \cdot} \mid j = 1, \dots, m\}$ of positive semidefinite matrices. Applying the matrix Chernoff inequality [320] on (5.56) yields

$$\Pr \left\{ \lambda_{\min} \left(\sum_{j \in \Lambda} \mathbf{X}_j \right) \geq \frac{s}{m} (1 - \delta) \right\} \geq 1 - r \left[\frac{e^{-\delta}}{(1 - \delta)^{1 - \delta}} \right]^{\frac{s}{m\alpha}}. \quad (5.57)$$

To show that $\mathbf{I}_{\Lambda, \cdot} \mathbf{G}$ is full rank, we must have $\frac{s}{m} (1 - \delta) > 0$. To achieve a better lower bound we set $(1 - \delta)$ to the smallest positive constant. This leads to the inequality

$$\Pr \left\{ \lambda_{\min} \left(\sum_{j \in \Lambda} \mathbf{X}_j \right) > 0 \right\} \geq 1 - r(0.3679)^{\frac{s}{m\alpha}}, \quad (5.58)$$

which completes the proof. The condition (5.54) is derived by isolating s in the inequality $1 - r(0.3679)^{\frac{s}{m\alpha}} \geq 0$. $\blacksquare$

(a) $\Psi = \{[\mathbf{R}^{(d)}, \mathbf{C}]\}_{d=1}^D$, where $\mathbf{R}$ is a uniform random orthonormal matrix and $\mathbf{C}$ is a DCT matrix.

(b) $\Psi = \{\mathbf{U}^{(3,d)} \otimes \mathbf{U}^{(2,d)} \otimes \mathbf{U}^{(1,d)}\}_{d=1}^D$ trained using the method described in Section 3.2.

(c) Ensemble of five K-SVD dictionaries trained on images.

Figure 5.6: Results of numerical simulations for Theorem 5.11. The signal size is $m = 12 \times 12 \times 3$. For the plots on the number of samples (s), we set $\tau = 5$. For the plots on the sparsity (τ), we set $s = m/2$. The dictionaries for (b) and (c) were trained on a set of 64 natural images.

5.8.4 Numerical Results

In Fig. 5.6 below, we report results for the numerical simulation of Theorem 5.11. The signal size is $m = 12 \times 12 \times 3$, which corresponds to patches of size 12×12 that are extracted from color images. Three types of dictionary ensembles are considered. Figure 5.6a reports results for an ensemble of concatenated dictionaries $\Psi = \{[\mathbf{R}^{(d)}, \mathbf{C}]\}_{d=1}^D$, where $\mathbf{R}$ is a uniform random orthonormal matrix and $\mathbf{C}$ is a DCT matrix. Figure 5.6b uses an MDE trained with the method described in Section 3.2 and Fig. 5.6c uses an ensemble of five K-SVD dictionaries trained on a set of natural images. To calculate the probability of uniqueness using (5.55), 10^5 trials are performed, where in each trial we randomly construct the support sets I and J , as well as the spike ensemble $\mathbf{I}_{\Lambda, \cdot}$. Moreover, two dictionaries are chosen at random from Ψ in each trial. The plots in Fig. 5.6 present the best case, worst case, and the median of trials.

5.9 Summary and Future Work

In this chapter, we introduced compressed sensing, a modern approach for sampling and reconstruction of sparse signals. It was shown that a sparse signal can be reconstructed from what appears to be highly inadequate nonadaptive samples. A number of important metrics that are used for evaluating the conditions of exact recovery were discussed. Using these metrics, as well as signal and measurement parameters, we formulated the conditions for exact recovery of a sparse signal. The recovery metrics we introduced in this chapter constitute a small number of seminal works in compressed sensing. There exists a large number of improvements over the metrics we described, as well as other types of analyses that introduce new metrics. Moreover, we reviewed the OMP algorithm, which is considered to provide a good balance between computational complexity and accuracy in sparse signal recovery.

A key contribution of this thesis, i.e. exact recovery conditions for OMP, was presented. In particular, a lower bound probability for successful support recovery of a sparse signal with additive white Gaussian noise was formulated. We also showed that by using an upper bound for the recovery error, one can obtain simplified formulas that reveal the effect of important signal parameters such as dynamic range and signal to noise ratio on the probability of support recovery. Numerical simulations of the derived theoretical guarantees show a much closer match to empirical performance of OMP than previous work. Another key contribution of this thesis was presented in this chapter, namely the uniqueness conditions for the recovery of a point-sampled signal that is sparse under an ensemble of dictionaries. It was shown that, regardless of the type of dictionaries in the ensemble, it is

possible to find a unique solution. We formulated a lower bound on the number of samples, as well as a lower bound on the probability of successful recovery.

There are several venues for future work. With regards to the theoretical results on OMP, we believe that the probability bound can be improved to better match the empirical results. While the proposed probability bound is significantly more accurate than previous work, there exists a considerable gap between our theoretical results and the empirical performance of OMP. There exists crude approximations in the proof of Theorem 5.8 that can be improved, albeit at the cost of losing the simplicity of the analysis. Additionally, we only considered additive white Gaussian noise. In many applications, the noise model is more sophisticated. For instance, the noise in an image obtained from a DLSR camera is a combination of several noise types such as read-out noise, photon shot noise, etc. An interesting direction for future research is to extend the results of Paper F to include different types of measurement noise. In Section 6.5, we will discuss other possible directions for future work with regards to utilizing the theoretical results of Paper F and Paper G in practice.

Theorem 5.11 presented a novel analysis for the uniqueness conditions of signals that are sparse in an ensemble of dictionaries. The current formulation relies on 1D signals (vectors). Hence, a multidimensional dictionary ensemble is converted to an equivalent 1D ensemble using (2.10) prior to utilizing the theorem. We believe that a new theoretical analysis that is tailored for an ensemble of multidimensional dictionaries is an interesting venue for future work. For instance, consider the following compressed sensing problem

$$\min_{\hat{\mathbf{S}}, \{\mathbf{U}^{(1,d)}, \dots, \mathbf{U}^{(n,d)}\}_{d \in \Psi}} \|\hat{\mathbf{S}}\|_0 \quad \text{s.t.} \quad \|\mathbf{Y} - \mathbf{S} \times_1 \mathbf{\Phi}^{(1)} \mathbf{U}^{(1,d)} \times_2 \dots \times_n \mathbf{\Phi}^{(n)} \mathbf{U}^{(n,d)}\|_2^2 \leq \epsilon, \quad (5.59)$$

where $\Psi = \{\mathbf{U}^{(1,d)}, \dots, \mathbf{U}^{(n,d)}\}_{d=1}^D$. Problems such as (5.59) have not been considered yet and there are several unsolved problems. Training an ensemble of dictionaries that enables sparse representation and performs well in equation (5.59) is an open problem. Another one is to derive performance guaranteed for (5.59) given an ensemble. Can the uniqueness analysis in Theorem 5.11 be adapted to (5.59) without the need for converting the problem to a 1D equivalent form? Note that Theorem 5.11 is valid for ensembles of any dimensionality when converted to the equivalent 1D form. However, it may be possible to derive a more accurate probability bound that is tailored for multidimensional ensembles.

Another direction for future research is to extend Theorem 5.11 to noisy signals. This is particularly important since Theorem 5.11 can be used for designing imaging systems such as light field cameras, which are prone to measurement noise. Moreover, we can include the effect of the representation error in our analysis.

Recall that the visual data are typically compressible and not sparse. Converting a compressible signal to a sparse signal introduces error (which is a part of lossy compression techniques we described in Chapter 3). In Theorem 5.11 we assumed that the signal is sparse in all the dictionaries, which may not be true for some signals in practice. Therefore, including the effect of noise and representation error can greatly improve the applicability of Theorem 5.11 in practice. The combined effect of the measurement noise and the representation error can be formulated using an upper bound on the total error of reconstruction.

Compressed Sensing in Graphics and Imaging

So far we have considered theoretical aspects of compressed sensing. Indeed the main goal of this field is to derive conditions under which one can exactly recover a sparse signal from what appears to be incomplete measurements. At the same time, because compressed sensing deals with the fundamental problem of sampling and reconstruction, it has found numerous applications in substantially different fields of science and engineering. While this thesis focuses on the applications of compressed sensing in computer graphics and image processing, the following subjects exemplify the diversity of research fields that use compressed sensing: wireless sensor networks [321, 322], face/gesture/speech recognition [63, 323, 324, 325], biomedical applications [326, 327], radar imaging [328, 329], integrated circuits [330, 331, 332], audio source separation [333, 334], and many more.

One of the early applications of compressed sensing was in imaging, specifically the “single pixel camera” [83]. Figure 6.1 demonstrates a block diagram of a single pixel camera. This imaging system consists of an array of digital micro-mirror devices (DMD) that can be individually addressed. A random number generator creates a random pattern on the DMD array for each shot. The incident light from the scene is reflected off of the random DMD array and integrated at a single photo diode. An analog to digital converter then creates a bit stream which is fed to an ℓ_1 reconstruction algorithm. The original work reports acceptable image quality from 1300 measurements, which is 50 times less than the Nyquist rate. The single pixel camera has been extended to accommodate active illumination [335], multispectral

Figure 6.1: Block diagram of a single pixel camera.

imaging [336, 337], video capture [338], microscopy [339], and terahertz imaging [340]. Another important application of compressed sensing in imaging is MRI, which was first introduced in [26]. Moreover, compressed sensing has been used for functional MRI [341, 342, 343, 344], and dynamic MRI [88, 345, 346, 347, 348, 349]. For a review of compressed sensing algorithms that can be used in dynamic MRI see [350].

Inverse problems in image processing can also be formulated using compressed sensing. Assume that an image is multiplied by a linear operator Φ which degrades the image. The operator Φ may add noise, blur, or subsample the image. The inverse of these operations are called denoising, deblurring, and super-resolution, respectively. Note that Φ can also be a combination of different operators. If the image is sparse, then the process of inverting the operator Φ to get back the original image is equivalent to solving (5.4) or (5.5). Compressed sensing has been used extensively for denoising [351, 352, 353, 354], deblurring [355, 356, 357, 358, 359], and super-resolution [360, 361, 362, 363].

To the best of the author’s knowledge, compressed sensing was first used in graphics for the acquisition of light transport of a scene [86]. The authors propose to project random illumination patterns onto a scene and perform measurements using a camera. The measurements are then used to reconstruct the light transport matrix using a modified version of ROMP [303]. Displays and projection systems have also utilized compressed sensing [364, 365]. Wang et al. [352] introduced a compressed sensing method for separating noise and features from polygonal meshes. Hue et al. [366] have used compressed sensing for addressing the many-lights problem in rendering. By taking a few measurements from the lighting matrix (that contains illumination information from virtual point lights to points in a scene), the authors show that a faithful recovery of the full matrix is possible. Unfortunately there are

only a few examples of the application of compressed sensing in computer graphics. Given the success of compressed sensing in vastly different fields of science and engineering, we believe that there exists many opportunities in computer graphics to be explored.

6.1 Outline and Contributions

This chapter will present the contributions of this thesis regarding the utilization of compressed sensing for solving various problems in graphics and imaging. Particularly, in Section 6.2, and also in Paper E, we present a light field camera design that utilizes compressed sensing to capture high resolution, high quality, light fields using a single sensor. Reconstruction results from the proposed algorithm show significant improvements over the prior work. Next, we show in Section 6.3 that compressed sensing can be used to accelerate photorealistic rendering techniques, which is the topic of discussion in Paper C. This is achieved by rendering a subset of pixel samples and reconstructing the remaining samples using compressed sensing. The results show that 60% of the pixels is adequate for a faithful recovery of a photorealistic rendered image. We also discuss how AMDE, introduced in Section 3.3 and Paper B, can be used for efficient compressed sensing of visual data regardless of the dimensionality (see Section 6.4). While we discuss selected examples such as light fields, light field videos, and BRDFs, the proposed framework can be used for the measurement of various types of visual data such as BTF, SVBRDF, multispectral light field videos, etc.

6.2 Light Field Imaging

Light field imaging has attracted a lot of attention over the past two decades. Numerous designs have been proposed which range from hand-held devices to large-scale stationary systems. In this section we describe the design of a multi-shot single-sensor light field camera that can be implemented in a small hand-held form factor. Before presenting the method in Section 6.2.2, we review relevant prior works in Section 6.2.1. The simulation results of the proposed light field camera will be discussed in Section 6.2.3.

6.2.1 Prior Art

One of the first attempts at capturing high quality light fields was with multi-sensor systems, also known as camera arrays [367]. By arranging several digital cameras on a regular grid, as shown in Fig. 6.2a, and triggering them at the same time, we obtain a set of images. Knowing the location and the parameters of the cameras, acquired images are reprojected to construct a light field. One can increase the

(a) Camera array [367]

(b) Microlens array

(c) Lytro Illum

(d) Mechanical arm [368]

(e) Angle sensitive pixels [369]

Figure 6.2: Light field capturing systems.

angular resolution by adding more cameras; hence, the system is scalable. Indeed high resolution light fields and light field videos can be captured using such systems. However, the cost and size of these devices limit their usability. In order to reduce the cost of multi-sensor devices, it is possible to mount a single camera on a mechanical arm, as shown in Fig. 6.2d, for capturing high resolution light fields [368, 370, 371]. However, these light field imaging systems can only be used for static scenes.

A popular and relatively cheap method for capturing light fields is through the utilization of microlens arrays [372, 373, 374]. Figure 6.2b shows a diagram of a microlens array light field camera. A large array of small lenses are placed in front of the sensor. The pixels corresponding to a microlens, say an $a_1 \times a_2$ grid, define the incoming radiance from $a_1 \times a_2$ directions. As a result, the spatial resolution is defined by the number of microlenses, and the angular resolution is defined by the number of pixels corresponding to a microlens. In fact, a microlens array sacrifices

spatial resolution of the sensor for angular resolution. Due to the fact that the microlens light field cameras are small in size and simple to implement, they were the first to be commercialized (see Fig. 6.2c). However, existing cameras have a low spatial resolution. Recently, a new sensor technology was introduced, namely an array of *angle sensitive pixels* (ASP) [375, 376]. Each ASP is tuned to have a predefined angular response by utilizing the Talbot effect. Using computational photography, it is possible to obtain a light field from an array of ASPs [369].

Another method for single sensor light field photography is *coded aperture* [377], where a randomly transparent non-refractive mask is placed on the aperture plane. In this way, each aperture region corresponding to a mask cell transmits a view of the scene modulated by the value of the mask cell. Hence the sensor integrates randomly modulated views of the light field. Knowing the transmittance values of the mask, it is possible to reconstruct the light field views. In [378], a programmable mask is placed at the aperture plane. This enables taking multiple shots while changing the pattern of the mask, hence increasing the number of samples and improving the image quality. However, this comes at the cost of long exposure times and a large number of shots. A dual mask light field camera was proposed in [379]. While the method increases spatial resolution, the utilization of two masks severely reduces the light transmission to the sensor. Babacan et al. [85] utilize compressed sensing to solve the inverse problem of obtaining a light field from coded aperture measurements.

More recently, Marwah et al. [17] propose a compressive light field camera where the mask is placed at a small distance, d_m , from the sensor. Let the function $l(\mathbf{r}_i, \mathbf{t}_j, \mathbf{u}_\alpha, \mathbf{v}_\beta)$ be a discrete two plane light field parametrization defined by a pair of locations on the image plane, $(\mathbf{r}_i, \mathbf{t}_j)$, and the aperture plane, $(\mathbf{u}_\alpha, \mathbf{v}_\beta)$. Then the image formed on the sensor, denoted $y(\mathbf{r}_i, \mathbf{t}_j)$, is obtained by

$$y(\mathbf{r}_i, \mathbf{t}_j) = \sum_{\alpha=1}^{|\mathbf{u}|} \sum_{\beta=1}^{|\mathbf{v}|} f(\mathbf{r}_i + \gamma(\mathbf{u}_\alpha - \mathbf{r}_i), \mathbf{t}_j + \gamma(\mathbf{v}_\beta - \mathbf{t}_j)) l(\mathbf{r}_i, \mathbf{t}_j, \mathbf{u}_\alpha, \mathbf{v}_\beta), \quad (6.1)$$

where $f(\cdot)$ defines the random mask, and $\gamma = d_m/d_a$, where d_a is the distance from the sensor to the aperture plane. Equation (6.1) is equivalent to

$$\mathbf{y} = \begin{bmatrix} \Phi^{(1)} & \Phi^{(2)} & \dots & \Phi^{(\nu)} \end{bmatrix} \begin{bmatrix} \mathbf{x}^{(1)} \\ \mathbf{x}^{(2)} \\ \vdots \\ \mathbf{x}^{(\nu)} \end{bmatrix}, \quad (6.2)$$

where $\nu = |\mathbf{u}||\mathbf{v}|$ and $\Phi^{(i)} \in \mathbb{R}^{\omega \times \omega}$, $\omega = |\mathbf{r}||\mathbf{t}|$, is a diagonal matrix that contains sheared mask values from $f(\cdot)$. Moreover, $\mathbf{x}^{(i)} \in \mathbb{R}^\omega$ contains a light field view image. By assuming sparsity of the light field in the K-SVD dictionary, the authors

solve (6.2) using an ℓ_1 program. Moreover, experiments conducted with binary, Gaussian, and optimized masks, show the superiority of optimized masks.

Figure 6.3: Multi-shot single-sensor light field camera with a color coded mask. For each shot, we move the mask randomly to create a new mask pattern.

6.2.2 Multi-shot Single-sensor Light Field Camera

Paper E presents a simple extension of [17] that substantially improves image quality. Instead of placing a monochrome mask, we use a color coded mask. This is shown in Figure 6.3. Moreover, we propose to take a small number of shots, where for each shot the mask (or the sensor) is moved randomly using e.g. a piezo motor. The movement of the mask creates new random modulation of the light field, and hence increasing the number of measurements, as well as reducing the coherence of measurement. Let s be the number shots, then the new measurement model is formulated as follows

$$\underbrace{\begin{bmatrix} \mathbf{y}^{(R,1)} \\ \mathbf{y}^{(G,1)} \\ \mathbf{y}^{(B,1)} \\ \mathbf{y}^{(R,2)} \\ \mathbf{y}^{(G,2)} \\ \mathbf{y}^{(B,2)} \\ \vdots \\ \mathbf{y}^{(R,s)} \\ \mathbf{y}^{(G,s)} \\ \mathbf{y}^{(B,s)} \end{bmatrix}}_{\triangleq \mathbf{y} \in \mathbb{R}^{3\omega s}} = \underbrace{\begin{bmatrix} \Phi^{(1,R,1)} & \dots & \Phi^{(\nu,R,1)} & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & \Phi^{(1,G,1)} & \dots & \Phi^{(\nu,G,1)} & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & \dots & 0 & \Phi^{(1,B,1)} & \dots & \Phi^{(\nu,B,1)} \\ \Phi^{(1,R,2)} & \dots & \Phi^{(\nu,R,2)} & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & \Phi^{(1,G,2)} & \dots & \Phi^{(\nu,G,2)} & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & \dots & 0 & \Phi^{(1,B,2)} & \dots & \Phi^{(\nu,B,2)} \\ \vdots & & & & & & & & \\ \Phi^{(1,R,s)} & \dots & \Phi^{(\nu,R,s)} & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & \Phi^{(1,G,s)} & \dots & \Phi^{(\nu,G,s)} & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & \dots & 0 & \Phi^{(1,B,s)} & \dots & \Phi^{(\nu,B,s)} \end{bmatrix}}_{\triangleq \Phi \in \mathbb{R}^{3\omega s \times 3\omega \nu}} \underbrace{\begin{bmatrix} \mathbf{x}^{(1,R)} \\ \vdots \\ \mathbf{x}^{(\nu,R)} \\ \mathbf{x}^{(1,G)} \\ \vdots \\ \mathbf{x}^{(\nu,G)} \\ \mathbf{x}^{(1,B)} \\ \vdots \\ \mathbf{x}^{(\nu,B)} \end{bmatrix}}_{\triangleq \mathbf{x} \in \mathbb{R}^{3\omega \nu}}. \quad (6.3)$$

Moreover, we propose to use spatial sampling to further decrease the coherence of measurements. The number of samples for spatial sampling is user-defined and provides a trade-off between the quality and the speed of reconstruction. This is useful since the measurement model in (6.3) has a fixed number of measurements,

Figure 6.4: Visual quality comparison between Paper E and Marwah et al. [17] with non-overlapping and overlapping patches using five shots.

i.e. the number of rows in Φ is fixed. Because the total number of samples is directly proportional to the reconstruction time, one can control the speed of reconstruction using the spatial sampling based on the hardware used in the camera. Let $\mathbf{D}$ be an overcomplete dictionary and $\mathbf{I}_{\Lambda_{\cdot}}$ be a spike ensemble. Then, the reconstruction of a light field captured using the measurement model in (6.3) is achieved by solving

$$\min_{\mathbf{s}} \|\mathbf{s}\|_1 \quad s.t. \quad \|\mathbf{y} - \mathbf{I}_{\Lambda_{\cdot}} \Phi \mathbf{D} \mathbf{s}\|_2^2. \quad (6.4)$$

6.2.3 Implementation and Results

We train a 1.5-times overcomplete dictionary on the same training set used in [17] and using the online dictionary learning method [164]. The light fields in the dataset have an angular resolution of 5×5 . The spatial patch size of the light field was set to 9×9 , therefore giving us data points with dimensionality $m_1 = 9$, $m_2 = 9$, $m_3 = 5$, $m_4 = 5$, $m_5 = 3$. Moreover, we used SL0 [82] for solving (6.4). Simulation results for the proposed multi-shot single-sensor light field camera are summarized in Fig. 6.4, and a physical implementation is left for future work. Since the hardware implementation of a monochrome mask [17, 377, 378], and random movements of a mask [203] has been done before, we believe that a physical implementation of the proposed light field camera is feasible. Moreover, note that we use uniformly random values for diagonal matrices in (6.3), while the results we report for Marwah et al. [17] use an optimized set of values. And yet our method significantly outperforms the approach proposed in [17].

Figure 6.5: Results of Paper E in comparison to [380] for a natural light field captured using the Lytro Illum as a part of the Stanford light field archive (<http://lightfields.stanford.edu/>).

In a parallel work to Paper E, Nabati et al. [380] propose a single-shot light field camera using a color coded mask. Instead of using compressed sensing, the authors utilize Convolutional Neural Networks (CNN) for light field reconstruction. The sensing model in [380] is slightly different than (6.3). In particular, all the color

Figure 6.6: Results of Paper E for a natural light field captured using Lytro Illum as a part of the Stanford light field archive (<http://lightfields.stanford.edu/>).

channels are measured into a single measurement. The measurement model in (6.3) using combined color sensing becomes

$$\underbrace{\begin{bmatrix} \mathbf{y}^{(1)} \\ \mathbf{y}^{(2)} \\ \vdots \\ \mathbf{y}^{(s)} \end{bmatrix}}_{\triangleq \mathbf{y} \in \mathbb{R}^{\omega \times \omega \times s}} = \underbrace{\begin{bmatrix} \Phi^{(1,R,1)} & \dots & \Phi^{(\nu,R,1)} & \Phi^{(1,G,1)} & \dots & \Phi^{(\nu,G,1)} & \Phi^{(1,B,1)} & \dots & \Phi^{(\nu,B,1)} \\ \Phi^{(1,R,1)} & \dots & \Phi^{(\nu,R,2)} & \Phi^{(1,G,2)} & \dots & \Phi^{(\nu,G,2)} & \Phi^{(1,B,2)} & \dots & \Phi^{(\nu,B,2)} \\ \vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ \Phi^{(1,R,s)} & \dots & \Phi^{(\nu,R,s)} & \Phi^{(1,G,s)} & \dots & \Phi^{(\nu,G,s)} & \Phi^{(1,B,s)} & \dots & \Phi^{(\nu,B,s)} \end{bmatrix}}_{\triangleq \Phi \in \mathbb{R}^{\omega \times s \times 3 \times \omega \times \nu}} \mathbf{x}. \quad (6.5)$$

We compare our results with that of [380] in Fig. 6.5 for a natural light field captured using the Lytro Illum, where the measurement model is according to (6.5). Using one shot, our method achieves a slightly higher PSNR than [380]. However, a visual comparison shows that our method reproduces colors more accurately and is sharper in general. With two shots, our method achieves 5.72db over [380] in PSNR. It should be noted that it is unclear if [380] can be extended for multi-shot light field capture since the mask is used during training. Using 5 shots, we achieve

near-perfect reconstruction and the PSNR gap with [380] becomes 11.68dB. In Fig. 6.6, we present another example of a natural light field reconstructed using the proposed light field camera of Paper E.

6.3 Accelerating Photorealistic Rendering

Photorealistic rendering has been a longstanding goal in computer graphics. With the advent of ray tracing [249, 250] and accurate surface reflectance modeling [381, 382], a large number of rendering techniques have emerged to progressively improve photorealism and computational cost of rendered images. Some examples of rendering techniques are: bidirectional path tracing [383, 384], photon mapping [385, 386, 387], and metropolis light transport [388, 389, 390, 391]. While the advances in rendering techniques and the ever increasing computational power have enabled a steady increase in photorealism of rendered images, photorealistic rendering of complex scenes still requires anywhere between minutes to hours, or even days of computing. We expect this problem to persist in the future since more and more physically accurate models for rendering will be utilized to improve photorealism. In this section, we present an image-based method that can be utilized on top of any rendering technique to accelerate the rendering process. We start by discussing a number of relevant works on the topic of accelerating the rendering process in sections 6.3.1 and 6.3.2. Our proposed method in Paper C will be described briefly in Section 6.3.3, followed by the results in Section 6.3.4.

6.3.1 Prior Art

Due to the high dimensionality of the rendering problem, Monte Carlo methods have been a key mathematical tool in solving the rendering equation. It is well-known that the error of a Monte Carlo estimator goes to zero, almost surely, as the number of samples goes to infinity. Due to the limited amount of computational power, noise or variance in the produced images are a part of photorealistic rendering techniques. Hence, rendering methods are typically evaluated based on the quality of the images produced versus the required number of samples. There exists several methods for improving the efficiency of rendering techniques given a fixed budget of samples. One method is to adaptively sample the space of parameters. In other words, we can distribute the fixed number of samples to parameters that contribute more to the quality of the rendered image. While this approach reduces noise, it can introduce bias in the estimator. A comprehensive review of adaptive sampling methods is presented in [392]. A different approach is to render using a small number of samples and then denoise the resulting estimated image [393, 394, 395, 396, 397]. An advantage of this method is that it is independent of the rendering algorithm and can be applied on top of adaptive sampling techniques. Another image-based

technique is to render a subset of pixels, or a subset of intra-pixel samples, and then use image processing techniques to estimate the missing samples, which will be described in the following section.

6.3.2 Compressive Rendering

Darabi and Sen [398] proposed *compressive rendering*, an algorithm for reconstructing an incomplete rendered image from random non-adaptive samples. The authors use wavelets as the sparsifying basis and a spike ensemble as the measurement matrix. It can be shown that the wavelet basis is coherent with the spike ensemble, i.e. the value of $\mu([\mathbf{I}_{\Lambda}, \mathbf{W}])$ is relatively high (see Definition 5.5), where $\mathbf{W}$ is the wavelet basis. In sections 5.5.1 and 5.7 we showed that a large value of mutual coherence reduces the probability of successful sparse signal recovery. To address the problem of coherence, the authors assume that there exists a blurred version of the original image that is sparse under the wavelet basis and can be sharpened to get the original image. Hence, the measurement model is extended with a sharpening filter that recovers the original image. This model cannot be mathematically verified since there is no guarantee that a blurred image can be recovered, and when combined with a wavelet basis, the recovery problem becomes more difficult. Moreover, numerical results for the proposed method in [398] are inferior to a linear interpolation method based on Delaunay triangulation (see Paper C). Moreover, due to the filtering procedure, one has to pad the rendered image, which leads to artifacts along the outer edges of the image. Without padding, only the central part of the reconstructed image is valid. Therefore, a significant portion of the rendered image, depending on how large the filter is, should be discarded. Indeed, if a suitable sparsifying basis that is incoherent with point sampling is used, there is no need to filter the input image.

6.3.3 Compressive Rendering Using MDE

In Paper C, we utilize the two dimensional variant of MDE introduced in Section 3.2 to solve the compressive rendering problem. A set of pixels are chosen uniformly at random for a ray tracer. After computing the pixel values, the incomplete image is given to a compressed sensing framework for the reconstruction of the original image by solving

$$\min_{\hat{\mathbf{s}}, (\mathbf{U}^{(2,d)} \otimes \mathbf{U}^{(1,d)}) \in \Psi} \|\hat{\mathbf{s}}\|_1 \quad \text{s.t.} \quad \|\mathbf{x} - \mathbf{I}_{\Lambda} \cdot (\mathbf{U}^{(2,d)} \otimes \mathbf{U}^{(1,d)}) \hat{\mathbf{s}}\|_2 \leq \epsilon, \quad (6.6)$$

where Ψ is an ensemble of two dimensional dictionaries trained on a set of natural and synthetic images. Note that the samples should not necessarily be full pixels. As long as sub-pixel samples lie on a regular grid, one can apply the proposed reconstruction algorithm on sub-pixel samples.

Input: Dictionary ensemble, $\Psi = \{\mathbf{U}^{(1,d)}, \mathbf{U}^{(2,d)}\}_{d=1}^D$, incomplete image, $\mathbf{F}$, index set of missing pixels, Γ , and number of iterations, I .
Variables: Image estimate, $\hat{\mathbf{F}}$, coefficients of the image estimate, $\mathbf{S}$.
Output: Index of best dictionary, a .

```

1  $\hat{\mathbf{F}} \leftarrow$  Perform DLI on  $\mathbf{F}$ ;
2 for  $i = 1$  to  $I$  do
3   Compute  $[\mathbf{S}, a] = \text{Test}(\hat{\mathbf{F}}, \Psi, \tau_t, \epsilon)$  using Algorithm 2;
4    $\hat{\mathbf{G}} \leftarrow \mathbf{U}^{(1,a)} \mathbf{S} (\mathbf{U}^{(2,a)})^T$ ;
5    $\hat{\mathbf{F}}_\Gamma \leftarrow \hat{\mathbf{G}}_\Gamma$ ;
6 end
```

Algorithm 6: Iterative algorithm for determining the best dictionary in the ensemble for compressive rendering.

As described in Section 5.8, since we do not know which dictionary leads to the most faithful recovery of the original image, one has to solve (6.6) for all the dictionaries in Ψ . Indeed this task is computationally expensive and departs us from the main goal, which is the acceleration of rendering techniques. In Paper C, we propose to utilize Algorithm 2 for determining the most sparse dictionary with the least reconstruction error prior to solving (6.6). By Theorem 5.11, we know that the number of samples is inversely proportional to sparsity; i.e. the more the number of nonzeros in a sparse signal, the more samples are required for accurate recovery. Since Algorithm 2 finds the dictionary leading to the most sparse representation, we expect the chosen dictionary to be compliant with the requirements of compressed sensing. However, the input to Algorithm 2 is the original signal, which is not available in compressive rendering because we only have the incomplete rendered image as input. To address this problem, we obtain an estimate of the original image using Delaunay Linear Interpolation (DLI) and use this estimate in Algorithm 2. Results show that even a crude approximation of the incomplete image can guide Algorithm 2 in finding the best dictionary. However, when the number of samples small, e.g. 20% of the total number of pixels, it is possible that the result of DLI becomes inadequate for determining the best dictionary. A simple extension to Paper C is to perform Algorithm 2 iteratively as outlined in Algorithm 6.

6.3.4 Implementation and Results

We trained an MDE on a dataset of 86 natural and synthetic images with parameters $K = 32$ and $\tau_l = 6$. Moreover, the image plane was divided into 12×12 patches. Since existing rendering libraries do not support independent rendering of a color

(a) Inset from a reference rendering

(b) Result of [398]: reconstruction from 20% of pixels, PSNR: 32.22dB

(c) Input: 20% of pixels rendered uniformly at random

(d) Result of Paper C: reconstruction from 20% of pixels, PSNR: 37.37dB

(e) Input: 60% of pixels rendered uniformly at random

(f) Result of Paper C: reconstruction from 60% of pixels, PSNR: 47.15dB

Figure 6.7: Reconstruction of incomplete rendered images from 20% and 60% of pixels using the compressed sensing framework of Paper C.

band, we do not include a color dimension in the data points. As a result, an MDE was trained for each color channel separately with data points of dimensionality $m_1 = 12$ and $m_2 = 12$. The three MDEs were combined to obtain one MDE.

Figure 6.8: Comparison of inpainting using AMDE with CCSC [399]. The input consists of 50% of pixels, i.e. the sampling ratio is 0.5.

Selected results of applying our method for reconstructing photorealistic renderings are presented in Fig. 6.7 and we refer the reader to Paper C for a thorough comparative set of examples. The reference image shown in Fig. 6.7 is an inset from a photorealistic rendering obtained with the *LuxRender* library. The results show a clear advantage of our method compared to [398]. Moreover, we see that with 60% of pixels, the reconstructed image is almost indistinguishable from a reference rendering.

Apart from the superior performance of the proposed compressed sensing framework for rendering, Paper C shows that the method outperforms state-of-the-art inpainting algorithms by a large margin. Specifically, comparisons were made with low rank tensor completion [400], steering kernel regression [401], piecewise linear estimators [402], nonparametric Bayesian dictionary learning [403], and K-SVD. Since the publication of Paper C, several methods have been proposed for image inpainting from random measurements. Recently, Convolutional Sparse Coding (CSC) algorithms have attracted a lot of attention for solving inverse problems in image processing. The CSC algorithm was introduced in [404], and several methods have been proposed for improving the efficiency of the algorithm [399, 405, 406, 407, 408]. In Fig. 6.8, we compare the result of using an AMDE for inpainting with Consensus-CSC (CCSC) [399], which to the best of author's knowledge is the current state-of-the-art method. Compressed sensing using AMDE achieves more than 3dB in PSNR over CCSC while being about an order of magnitude faster.

Figure 6.9: Visual quality results for 1D compressed sensing of a natural light field (*Lego Knights*) and a light field video (*Painter* [262]) using AMDE for different sampling ratios. We used SL0 [82] for solving (5.8).

6.4 Compressed Sensing of Visual Data

In sections 6.2 and 6.3, we considered two applications of compressed sensing in computer graphics and rendering. The proposed light field camera utilizes an overcomplete dictionary for sparse representation, while for the acceleration of ray tracing we used a 2D dictionary ensemble. As discussed in Chapter 3, AMDE provides superior performance in sparse representations. As a result, we expect improved reconstruction quality in various applications that utilize AMDE as a sparsifying basis in compressed sensing. In what follows, we consider two additional applications of compressed sensing in graphics that facilitate high dimensional visual data measurements using AMDE. In particular, we discuss light field reconstruction from random point sampling in Section 6.4.1 and compressive BRDF capture in Section 6.4.2. Indeed, since AMDE can be trained on any n D visual data set, these two applications are merely examples of possibilities for utilizing AMDE in compressed sensing.

6.4.1 Light Field and Light Field Video Completion

In this section, we discuss the problem of reconstructing a light field, or a light field video, from a small subset of measurements that are acquired by uniform sampling at random. This is different from the proposed light field camera in

Figure 6.10: Visual results for 5D compressed sensing of *Lego Knights* using AMDE. We used GTCS [160] for solving (5.6)

Section 6.2 for two reasons. First, here the measurement matrix is a spike ensemble, see Definition 5.7, and does not possess the special structure of (6.3). Moreover, the sparsifying basis used in this section is AMDE. Two datasets are considered here: *Lego Knights* from the Stanford Lego Gantry dataset and *Painter* from [262], where the former is a light field and the latter is a light field video dataset. For *Lego Knights*, we create data points of size $m_1 = 5$, $m_2 = 5$, $m_3 = 8$, $m_4 = 8$, and $m_5 = 3$. For *Painter*, we use $m_1 = 10$, $m_2 = 10$, $m_3 = 4$, $m_4 = 4$, $m_5 = 3$, and $m_6 = 4$, where the last dimension is the number of consecutive frames used in a data point. An AMDE is learned for each of these applications on a training set that excludes *Lego Knights* and *Painter*. Each AMDE consists of 128 multidimensional dictionaries. Furthermore, we use the procedure described in Section 6.3 for determining the most suitable dictionary. The selected dictionary, together with the incomplete data points and the spike ensemble are used to solve a 5D, or a 6D, variant of (5.8). We used SL0 [82] for this purpose. The results are summarized in Fig. 6.9.

It should be noted that although the AMDEs for light fields and light field videos are 5D and 6D, respectively, we use the vectorized form of AMDE to solve the compressed sensing problem. A more versatile method is to solve (5.6) instead. In this way, each dimension of the light field (video) is sampled independently. However, a robust multidimensional ℓ_0 or ℓ_1 solver that matches the performance of a vectorized solver such as SL0 does not exist. Indeed since AMDE enables a more sparse representation than a 1D dictionary such as K-SVD (see e.g. Fig. 4.5), the compressed sensing performance is expected to be improved using an effective multidimensional solver. As an example, we used GTCS [160] to solve (5.6). To the best of author’s knowledge, GTCS is the state of the art solver for multidimensional compressed sensing. However, we found that the results of this solver is vastly inferior to SL0, see Fig. 6.10. We have left the design of an efficient multidimensional solver for future work.

6.4.2 Compressive BRDF capture

The Bidirectional Radiance Distribution Function (BRDF) [410] describes the reflectance properties of a point on a surface. In essence, a BRDF is a function that describes the appearance of a point under varying illumination from different

Figure 6.11: Gonioreflectometer for BRDF measurement. Two mechanical arms, one holding a camera and the other holding a light source, cover the entire hemisphere above a sample material patch. The exemplar above was developed in collaboration between Rayspace AB and Linköping University. Implementation details can be found in [409].

Figure 6.12: Compressive BRDF capture for two BRDFs from the MERL database. The peak value for PSNR of a reconstructed BRDF is set to the largest value in the original BRDF.

vantage points. Indeed, an accurate description of reflectance properties is a key component of photorealistic rendering. The BRDF is represented by a 4D scalar-valued function $b(\phi_i, \theta_i, \phi_o, \theta_o)$, where (ϕ_i, θ_i) is the direction towards the light source in the spherical coordinates and (ϕ_o, θ_o) is the view direction. The value of the function describes the ratio of the reflected radiance along (ϕ_o, θ_o) to the irradiance arriving from (ϕ_i, θ_i) . In this section, we focus on measured BRDFs, which are a key component in deriving physically based [38, 411, 412], data-driven [413, 414], and empirical [415, 416] BRDF models. A measured BRDF is merely

Figure 6.13: Image based lighting of the reconstructed *gold-metallic-paint3* BRDF for different sampling ratios. The PSNR is calculated over the rendered image. The PSNR of the BRDF itself is reported in Fig. 6.12a.

a discretization of the function $b(\phi_i, \theta_i, \phi_o, \theta_o)$, where the surface reflectance is measured for a set of (discrete) values of the parameters. As a result, a measured BRDF is a 4D tensor, or a 5D tensor including a wavelength dimension.

There are many approaches for measuring the BRDF of a point on a surface. A common method for BRDF acquisition, and possibly the most accurate one, is to use a *gonioreflectometer*, shown in Fig. 6.11. The system consists of two mechanical arms, one holding a camera and the other holding a light source. The movements of each mechanical arm covers the entire hemisphere centered at the sample point, hence measuring the entire 5D BRDF domain. The accuracy of a gonioreflectometer comes at the cost of long scanning hours. In this section, we show that compressed sensing can be used for accelerating this process. To do this, a spike ensemble is used to sample a small portion of the view and light directions at random. For training the AMDE, we use 50 BRDFs from the MERL database [37].

We perform sampling and reconstruction using compressed sensing on two BRDFs

Figure 6.14: Image based lighting of the reconstructed *white-paint* BRDF for different sampling ratios. The PSNR is calculated over the rendered image. The PSNR of the BRDF itself is reported in Fig. 6.12b.

from the MERL database, namely *gold-metallic-paint3* and *white-paint*, which were not included in the training set. All the BRDFs were converted to a bivariate representation [417] prior to training or reconstruction. Moreover, the reconstruction was done by solving (5.8) using SL0. The reconstruction quality, in terms of PSNR, with respect to the sampling ratio is reported in Fig. 6.12. We used PBRT [15] to render the reconstructed BRDFs by utilizing image based lighting. The results, which are not published yet, are summarized in figures 6.13 and 6.14. We see that with 20% of BRDF samples, the rendered image is very close to the reference rendering, which was rendered using the full BRDF. With as low as 40% of total samples, a near-perfect rendering is achieved. Note that the samples are in the bivariate space. Therefore, our sampling ratio with regards to the full BRDF is significantly lower than the ratio reported here.

6.5 Summary and Future Work

In this chapter, we presented three applications in computer graphics and image processing that utilize the rich theoretical foundation of compressed sensing. First, a light field camera design was proposed that is simple to implement, yet provides great flexibility to capture high quality light fields without the requirement of multiple sensors. The simulation results of the proposed method show significant improvements over previous work. Equation (6.3) shows that the measurement matrix for a single-sensor compressive light field camera possesses certain structures and is not fully random. An exciting venue for research is to compute metrics for exact recovery such as mutual coherence, RIC, and ERC for the measurement matrix in (6.3). The author believes that it is possible to derive analytical formulas for a collection of recovery metrics given the structure present in the sensing matrix. In this way, one can use a rich collection of existing theoretical results for recovery conditions based on the aforementioned metrics.

The compressed sensing framework used for the light field camera is based on a variant of K-SVD [164], hence the light field is assumed to be a 1D signal. Given the superior results of the AMDE in compression, which is achieved by increasing sparsity, we expect improvements in the results of the light field camera with a multidimensional compressed sensing framework based on the AMDE. Since multidimensional sparse recovery algorithms have not matured yet, we have left the design of a multidimensional light field camera for future work. Indeed such an approach can offer important improvements. For instance, it is known that a light field exhibits different patterns of sparsity along its various dimensions. A light field may be sparse along its angular dimension but dense along the spatial dimension, or vice versa. As a result, it is beneficial to dedicate more samples along dimensions that are less sparse. This is only possible when a multidimensional compressed sensing is used (i.e. equation (5.6)).

The proposed light field camera in Section 6.2 can be extended for capturing High Dynamic Range (HDR) light fields and light field videos. There exists single-shot, single-sensor HDR imaging systems [418, 419, 420], where an HDR image is reconstructed from two horizontal interlaced ISO settings over the sensor (also known as dual-ISO HDR imaging). Such a reconstruction procedure can indeed be formulated using compressed sensing. Therefore, it is possible to combine the reconstruction of a dual-ISO HDR image and the measurement model of the proposed light field camera in (6.3); hence, achieving a single-shot HDR light field imaging system.

An interesting direction for future research is the utilization of Theorem 5.11 for guiding the design of compressive light field cameras that utilize an ensemble of dictionaries. Theorem 5.11 shows that the maximum row-norm of the equivalent

measurement matrix should be as small as possible. This requirement is equivalent to having an equivalent measurement matrix with relatively uniform values for row norms (assuming that the columns of the equivalent measurement matrix are normalized). There are two important factors that determine the equivalent measurement matrix. First, the design of the light field camera directly affects the structure of the measurement matrix. For instance, by changing the placement of the mask, one obtains a different measurement matrix. Second, the type of the dictionaries in the ensemble also affects the equivalent measurement matrix. Indeed if we know the properties of a “good” measurement matrix, as defined in Theorem 5.11, hardware implementations can be more cost effective and efficient. The theoretical analysis of existing light field capture systems using Theorem 5.11 is also left for future work. Of particular interest to the author is the design of an ensemble of n D overcomplete dictionaries and separable sensing operators that conform to the requirements of Theorem 5.11.

We discussed a method for accelerating the photorealistic rendering techniques. The proposed approach is based on rendering a subset of pixels and using a compressed sensing framework to reconstruct the remaining pixels. A sampling ratio of 60% shows production quality results using the proposed framework. Moreover, at a ratio of 20%, an acceptable image quality is achieved. An advantage of this method is that it can be applied to any ray tracing technique since we only consider image plane samples. Extension of the proposed compressive rendering to the full ray tracing pipeline is left for future work. In other words, it is possible to evaluate the reflection equation at each bounce using compressed sensing. We believe that this approach is feasible since several building blocks has already been discussed in this thesis. For instance, BRDFs are known to be very sparse objects in a suitable basis. Moreover, the reflectance equation at a point is expressed by the convolution of incoming radiance with the BRDF [45]. Since a convolution can be written as a matrix/vector multiplication, we can apply compressed sensing to determine the outgoing radiance.

In Chapter 3, we showed (empirically) that the ensemble of multidimensional dictionaries promotes sparsity. Hence, the author is interested in exploring and designing new approaches for efficient utilization of dictionaries ensembles for compressed sensing. Given that multidimensional compressed sensing and multidimensional overcomplete dictionary learning are unsolved problems, we believe that there exists several new research directions. Note that AMDE, introduced in Section 3.3, is an ensemble of multidimensional *orthonormal* dictionaries. Therefore, we expect significant improvements in compressed sensing performance for an ensemble of *overcomplete* multidimensional dictionaries. This is due to the fact that an overcomplete dictionary, compared to an orthonormal one, can produce sparse representations for a larger set of signals.

Concluding Remarks and Outlook

In this thesis, we presented methods for compression and compressed sensing of visual data. The foundation of these methods is sparse signal representation. In an interesting article, M. Elad [421] presents the progression of sparse representation techniques since the inception of the field, as well as dividing the future work into three major directions:

1. **Theoretical frontiers:** Improving existing theoretical guarantees in compressed sensing to better match empirical results.
2. **Model improvements:** Establishing new dictionary learning techniques that promote sparsity, while reducing the representation error.
3. **Applications:** Apply sparse representation and compressed sensing techniques to improve the quality, as well as efficiency, of compression and sampling (capturing) of visual data.

This thesis makes a number of contributions in all the three areas mentioned above. In what follows, we will summarize these contributions and state several unanswered research questions that can further push the boundaries of sparse representation techniques for visual data in the near future.

7.1 Theoretical Frontiers

This thesis presented two key contributions on advancing the theoretical frontiers of sparse representations and compressed sensing. First, we presented exact recovery conditions of the OMP algorithm for sparse signals contaminated with white Gaussian noise. The lower bound on the probability of success for OMP includes important (and easily computable) signal parameters, as well as dictionary properties. Additionally, we combined an upper bound on the representation error with the lower bound on the probability of success to derive new bounds that show the effect of signal parameters on the performance of OMP. Second, we derived uniqueness conditions for exact recovery of multidimensional sparse signals under an ensemble of multidimensional dictionaries (e.g. AMDE). This novel analysis shows the required conditions for a *suitable* multidimensional ensemble of dictionaries that enables efficient compressed sensing. Moreover, we derived the required number of samples and sparsity of signals, which are crucial in designing state of the art measurement techniques for visual data including light field cameras, as well as BRDF and BTF measurements devices. The author of this thesis believes that the results presented here have posed many new questions that can be subjects for future research. For instance,

- Theorem 5.8 was shown to significantly improve the previous results using mutual coherence for the recovery conditions of OMP. However, there is still a large gap between the theoretical and empirical results. Can the analysis be improved to have a better match between empirical and theoretical results?
- There exists multidimensional variants of OMP [102]. Can we extend the analysis in Paper F to accommodate these techniques?
- Theorem 5.11 presented theoretical guarantees for an ensemble of multidimensional dictionaries in a vectorized formulation, i.e. (5.46). Can we derive theoretical guarantees for an ensemble of multidimensional dictionaries in the native nD formulation (i.e. (5.59))?
- Using the theoretical results presented in this thesis, can we derive new performance guarantees for existing capturing mechanism of visual data such as BRDFs and BTFs?
- Can we derive optimal sensing matrices (according to various metrics in compressed sensing) that enable efficient BRDF capturing with the minimal number of samples?
- Can we derive closed form formulas for mutual coherence, RIP, or ERC of the single sensor light field camera with a color coded mask? And from that,

can we derive a lower bound on the number of shots that takes as input the maximum tolerable error in the reconstruction?

7.2 Model Improvements

Sparsity of a signal reduces the storage complexity, a concept that is the key building block of almost every compression method. Moreover, in Chapter 5 we showed that the sparsity of a signal is inversely proportional to the required number of samples for compressed sensing. We also discussed that visual data is typically compressible rather than sparse. Hence the induced sparsity in the representation relies on the quality of the dictionary. Therefore, a key challenge is to find optimal dictionaries that produce the most sparse representations with the least amount of error. More importantly, we need representations that are optimal for a large family of signals, what we called in this thesis the *representativeness* of the dictionary. Moreover, visual data are multidimensional objects that vary in dimensionality depending on the application and the way they are parameterized. Using these observations, dictionary learning methods were proposed that enable enhanced sparsity of multidimensional visual data. Specifically, we demonstrated the advantages of an ensemble of multidimensional dictionaries for compression in Chapter 3, and in chapters 5 and 6 for compressed sensing. In addition, we proposed methods to improve the quality of representation for multidimensional dictionaries by performing a sparsity-based pre-clustering of the training data. The resulting ensemble, called AMDE, is more powerful in the sparse coding of complex variations within, and among, data points extracted from visual datasets.

Improvements in sparsity were shown to enhance compression performance for various data sets such as SLF, light field, light field video, and appearance data. Given that the AMDE is multidimensional and expandable in the number of dictionaries, we believe that there exists numerous applications that can utilize AMDE for compression. Moreover, because of the superior performance of AMDE in sparse representations, as well as its flexibility on data dimensionality, any discrete multidimensional visual dataset can be efficiently compressed or captured using AMDE.

We found that there are numerous ways to improve the signal models we proposed in this thesis, as well as new directions for future research regarding the derivation of new models. Below we summarize a few unanswered questions for future research:

- Is it possible to train an ensemble of multidimensional overcomplete dictionaries? We know that overcomplete dictionaries, which initiated the field of sparse representations, are superior to orthogonal dictionaries. Moreover, we

showed in Chapter 3 that an ensemble of multidimensional orthogonal dictionaries is superior to a single overcomplete dictionary. Therefore, it is expected (by intuition) that an ensemble of multidimensional overcomplete dictionaries will significantly improve the signal model for sparse representation.

- The AMDE is trained by applying pre-clustering to the training data points, followed by ensemble training for each pre-cluster. Indeed an iterative approach can improve the quality of AMDE for sparse representation. It is left to be seen how much this can improve the performance of AMDE.
- Inspired by wavelets, can we derive a training method that iteratively applies pre-clustering and dictionary training in a multi-level fashion?
- Can we find methods for automatic tuning of the number of pre-clusters and dictionaries? This is indeed an open problem in most clustering and dictionary training algorithms.

7.3 Applications

We considered several applications for utilizing sparse representations of visual data in compression and compressed sensing. For compression, we used a variant of MDE for precomputed photorealistic rendering without any limitations on the complexity of materials or light sources. We also showed that AMDE is significantly superior to previous methods in compression of light field and light field video datasets. Moreover, we demonstrated that AMDE enables real time playback of high-resolution light field videos. To show the flexibility of AMDE, we also presented the compression of large-scale appearance data captured using a rotating camera rig. For compressed sensing, a design of a light field camera with a single sensor and a color coded mask was proposed which enables fast, high quality, and cheap light field capture. Moreover, we used ensemble of 2D dictionaries to accelerate photorealistic rendering techniques. And finally, we demonstrated that the ensemble of multidimensional dictionaries can be used for compressed sensing of a variety of datasets including light fields, light field videos, and BRDFs.

The possibilities of utilizing the proposed methods, and in general sparse representations, in computer graphics, imaging, and visualization are, figuratively, endless. There are only a few works on sparse representations and compressed sensing in computer graphics. Hence there exists numerous directions for future work, both by the extension of the methods discussed in this thesis, as well as their applications in new areas of computer graphics. For instance,

- We only discussed selected applications in computer graphics for compression and compressed sensing of visual data. Utilizing AMDE for compression of

Figure 7.1: Number of papers containing “sparse representation” in the title since 2006. For this chart we used Google Scholar and excluded patents. In 2017, the number of papers containing the phrase “sparse representation” was 11,800.

Figure 7.2: Number of papers containing “compressed sensing” in the title since 2006. For this chart we used Google Scholar and excluded patents. In 2017, the number of papers containing the phrase “compressed sensing” was 10,200.

BRDFs, multispectral light fields, BTFs, as well as scientific multidimensional data is left for future work. Similarly, compressed sensing of BTFs, and other types of multidimensional datasets are possible using an AMDE.

- It is interesting to see whether a complete reformulation of the rendering equation using compressed sensing is possible. After all, rendering is merely a sampling and reconstruction problem. Hence, intuitively it makes sense to use compressed sensing, which is a vastly superior replacement for the classical sampling techniques. Moreover, we demonstrated the compressed sensing performance for a variety of components used in rendering such

as BRDFs, images, and the sparsity present in the incoming and outgoing radiance distributions (i.e. the SLF).

- The precomputed rendering technique proposed in this thesis using SLFs (Section 4.2) can be straightforwardly extended to BTFs. In this way, the light sources can be dynamic during the real time rendering.
- Compressed sensing of visual data using AMDE can be extended for solving a variety of inverse problems such as denoising, deblurring, and super-resolution.
- Design of compressive measurement devices for BRDFs with minimal number of samples is also an interesting topic for future work. Existing methods are based on heuristics [422], which are difficult to evaluate with the limited amount of available measured BRDF datasets. Indeed this extends to other types of visual data such as BTFs.
- Capturing the light transport of a scene using the appearance measurement device introduced in Section 4.4 opens up many new research directions for real time photorealistic rendering of complex scenes. To do this, one can project the AMDE (or a variant of it) using an LED array onto the scene and use the cameras to capture the coefficients directly.
- In solving inverse problems in image processing, it is a common practice to create overlapping patches on an image. After reconstructing each patch, the results are combined (typically averaged) to form the reconstructed image. Can we utilize the theoretical results of Chapter 5 for patch-based image processing? In other words, is it possible to derive confidence measures for each reconstructed patch such that a weighted average of patches are used to form the reconstructed image? Indeed, this can be extended to higher dimensional visual data.

7.4 Final Remarks

While in this thesis we proposed several methods for sparse representation and compressed sensing, the contributions are dwarfed by the massively growing amount of research that is being carried out in these fields. In figures 7.1 and 7.2, we plot the number of papers published in peer-reviewed journals and conferences that contain the terms “sparse representation” and “compressed sensing” in their title, respectively. The number of papers that contain these terms inside the text is significantly higher; for instance, the term “compressed sensing” was used in 10,200 publications (excluding patents) in the year 2017 alone. Note that this excludes similar terms such as “compressive sensing” and “compressive sampling”. Indeed the figures show a growing interest in these fields, and most importantly, they

show an increasing number of research questions year by year. This thesis is no exception in this trend and we have presented more questions for future work than the questions we hopefully have answered by the contributions of this thesis. The author of this thesis finds the growing amount of opportunities in these fields, and especially their applications in computer graphics, to be extremely exciting.

Bibliography

- [1] E. Miandji, S. Hajisharif, and J. Unger, “A Unified Framework for Compression and Compressed Sensing of Light Fields and Light Field Videos,” *ACM Transactions on Graphics*, Provisionally accepted.
- [2] 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, Sept 2018.
- [3] E. Miandji[†], M. Emadi[†], and J. Unger, “OMP-based DOA Estimation Performance Analysis,” *Digital Signal Processing*, vol. 79, pp. 57–65, Aug 2018, [†] equal contributor.
- [4] 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, [†] equal contributor.
- [5] E. Miandji and J. Unger, “On Nonlocal Image Completion Using an Ensemble of Dictionaries,” in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, Sept 2016, pp. 2519–2523.
- [6] E. Miandji, J. Kronander, and J. Unger, “Compressive Image Reconstruction in Reduced Union of Subspaces,” *Computer Graphics Forum*, vol. 34, no. 2, pp. 33–44, May 2015.
- [7] 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*. ACM, 2013, pp. 24:1–24:4.
- [8] G. Baravdish, E. Miandji, and J. Unger, “GPU Accelerated Sparse Representation of Light Fields,” in *14th International Conference on Computer Vision Theory and Applications (VISAPP 2019)*, submitted. [page 57]
- [9] E. Miandji[†], M. Emadi[†], and J. Unger, “A Performance Guarantee for Orthogonal Matching Pursuit Using Mutual Coherence,” *Circuits, Systems, and Signal Processing*, vol. 37, no. 4, pp. 1562–1574, Apr 2018, [†] equal contributor.

- [10] J. Kronander, F. Banterle, A. Gardner, E. Miandji, and J. Unger, “Photo-realistic Rendering of Mixed Reality Scenes,” *Computer Graphics Forum*, vol. 34, no. 2, pp. 643–665, May 2015.
- [11] S. Mohseni, N. Zarei, E. Miandji, and G. Ardeshir, “Facial Expression Recognition Using Facial Graph,” in *Workshop on Face and Facial Expression Recognition from Real World Videos (ICPR 2014)*. Cham: Springer International Publishing, 2014, pp. 58–66.
- [12] S. Hajisharif, J. Kronander, E. Miandji, and J. Unger, “Real-time Image Based Lighting with Streaming HDR-light Probe Sequences,” in *SIGRAD 2012: Interactive Visual Analysis of Data*. Linköping University Electronic Press, Nov 2012.
- [13] E. Miandji, J. Kronander, and J. Unger, “Geometry Independent Surface Light Fields for Real Time Rendering of Precomputed Global Illumination,” in *SIGRAD 2011*. Linköping University Electronic Press, 2011.
- [14] E. Miandji, M. H. Sargazi Moghadam, F. F. Samavati, and M. Emadi, “Real-time Multi-band Synthesis of Ocean Water With New Iterative Up-sampling Technique,” *The Visual Computer*, vol. 25, no. 5, pp. 697–705, May 2009. [page 47]
- [15] M. Pharr and G. Humphreys, *Physically Based Rendering, Second Edition: From Theory To Implementation*, 2nd ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2010. [pages xiii, 51, and 107]
- [16] S. Lloyd, “Least squares quantization in PCM,” *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982. [pages xiii and 39]
- [17] K. Marwah, G. Wetzstein, Y. Bando, and R. Raskar, “Compressive Light Field Photography Using Overcomplete Dictionaries and Optimized Projections,” *ACM Transactions on Graphics*, vol. 32, no. 4, pp. 46:1–46:12, July 2013. [pages xv, 10, 93, 94, and 95]
- [18] 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 xv, 74, 75, and 76]
- [19] T. Hey, S. Tansley, and K. Tolle, *The Fourth Paradigm: Data-Intensive Scientific Discovery*. Microsoft Research, Oct 2009. [page 1]
- [20] 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, Aug 1996, pp. 43–54. [pages 1 and 27]
- [21] B. S. Wilburn, M. Smulski, H.-H. K. Lee, and M. A. Horowitz, “Light field video camera,” pp. 4674 – 4674 – 8, 2001. [page 1]
 - [22] G. D., G. G.C., G. A., D. C., and G. M., “BRDF Representation and Acquisition,” *Computer Graphics Forum*, vol. 35, no. 2, pp. 625–650, May 2016. [page 1]
 - [23] Z. Zhou, G. Chen, Y. Dong, D. Wipf, Y. Yu, J. Snyder, and X. Tong, “Sparse-as-possible SVBRDF Acquisition,” *ACM Transactions on Graphics*, vol. 35, no. 6, pp. 189:1–189:12, Nov 2016. [page 1]
 - [24] G. R. Arce, D. J. Brady, L. Carin, H. Arguello, and D. S. Kittle, “Compressive Coded Aperture Spectral Imaging: An Introduction,” *IEEE Signal Processing Magazine*, vol. 31, no. 1, pp. 105–115, Jan 2014. [page 1]
 - [25] K. J. Dana, B. van Ginneken, S. K. Nayar, and J. J. Koenderink, “Reflectance and Texture of Real-world Surfaces,” *ACM Transactions on Graphics*, vol. 18, no. 1, pp. 1–34, Jan 1999. [page 1]
 - [26] 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. [pages 1, 10, and 90]
 - [27] G. Wetzstein, D. Lanman, M. Hirsch, W. Heidrich, and R. Raskar, “Compressive Light Field Displays,” *IEEE Computer Graphics and Applications*, vol. 32, no. 5, pp. 6–11, Sept 2012. [page 2]
 - [28] G. Wetzstein, D. Lanman, M. Hirsch, and R. Raskar, “Tensor Displays: Compressive Light Field Synthesis Using Multilayer Displays with Directional Backlighting,” *ACM Transactions on Graphics*, vol. 31, no. 4, pp. 80:1–80:11, Jul 2012. [pages 2 and 32]
 - [29] S. Lee, C. Jang, S. Moon, J. Cho, and B. Lee, “Additive Light Field Displays: Realization of Augmented Reality with Holographic Optical Elements,” *ACM Transactions on Graphics*, vol. 35, no. 4, pp. 60:1–60:13, Jul 2016. [page 2]
 - [30] B. Koniaris, M. Kosek, D. Sinclair, and K. Mitchell, “Real-time Rendering with Compressed Animated Light Fields,” in *Proceedings of the 43rd Graphics Interface Conference*, ser. GI '17. Canadian Human-Computer Communications Society, 2017, pp. 33–40. [page 2]
 - [31] W.-C. Chen, J.-Y. Bouguet, M. H. Chu, and R. Grzeszczuk, “Light Field Mapping: Efficient Representation and Hardware Rendering of Surface Light

- Fields,” *ACM Transactions on Graphics*, vol. 21, no. 3, pp. 447–456, Jul 2002. [pages 2 and 47]
- [32] 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 2]
- [33] R. Ramamoorthi, M. Koudelka, and P. Belhumeur, “A Fourier theory for cast shadows,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 2, pp. 288–295, Feb 2005. [page 2]
- [34] 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. [pages 2, 32, and 34]
- [35] X. Sun, Q. Hou, Z. Ren, K. Zhou, and B. Guo, “Radiance Transfer Biclustering for Real-Time All-Frequency Byscale Rendering,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 1, pp. 64–73, Jan 2011. [page 2]
- [36] G. Müller, G. H. Bendels, and R. Klein, “Rapid Synchronous Acquisition of Geometry and Appearance of Cultural Heritage Artefacts,” in *Proceedings of the 6th International Conference on Virtual Reality, Archaeology and Intelligent Cultural Heritage*, ser. VAST’05. Eurographics Association, 2005, pp. 13–20. [page 2]
- [37] W. Matusik, H. Pfister, M. Brand, and L. McMillan, “A Data-driven Reflectance Model,” *ACM Transactions on Graphics*, vol. 22, no. 3, pp. 759–769, Jul 2003. [pages 2 and 106]
- [38] J. Löw, J. Kronander, A. Ynnerman, and J. Unger, “BRDF Models for Accurate and Efficient Rendering of Glossy Surfaces,” *ACM Transactions on Graphics*, vol. 31, no. 1, pp. 9:1–9:14, Feb 2012. [pages 2 and 105]
- [39] M. M. Bagher, C. Soler, and N. Holzschuch, “Accurate fitting of measured reflectances using a Shifted Gamma micro-facet distribution,” *Computer Graphics Forum*, vol. 31, no. 4, pp. 1509–1518, Jun 2012. [page 2]
- [40] R. Pacanowski, O. S. Celis, C. Schlick, X. Granier, P. Poulin, and A. Cuyt, “Rational BRDF,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 11, pp. 1824–1835, Nov 2012. [page 2]

- [41] A. Ngan, F. Durand, and W. Matusik, “Experimental Analysis of BRDF Models,” in *Proceedings of the Sixteenth Eurographics Conference on Rendering Techniques*, ser. EGSR ’05. Eurographics Association, 2005, pp. 117–126. [page 2]
- [42] D. L. Donoho, “Compressed sensing,” *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, Apr 2006. [pages 3 and 60]
- [43] B. Cabral, N. Max, and R. Springmeyer, “Bidirectional Reflection Functions from Surface Bump Maps,” *SIGGRAPH Computer Graphics*, vol. 21, no. 4, pp. 273–281, Aug 1987. [page 3]
- [44] P.-P. Sloan, J. Kautz, and J. Snyder, “Precomputed Radiance Transfer for Real-time Rendering in Dynamic, Low-frequency Lighting Environments,” in *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH ’02. ACM, 2002, pp. 527–536. [pages 3 and 45]
- [45] F. Durand, N. Holzschuch, C. Soler, E. Chan, and F. X. Sillion, “A Frequency Analysis of Light Transport,” *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 1115–1126, Jul 2005. [pages 3 and 109]
- [46] R. Ramamoorthi, D. Mahajan, and P. Belhumeur, “A First-order Analysis of Lighting, Shading, and Shadows,” *ACM Transactions on Graphics*, vol. 26, no. 1, Jan 2007. [page 3]
- [47] T. Malzbender, “Fourier Volume Rendering,” *ACM Transactions on Graphics*, vol. 12, no. 3, pp. 233–250, Jul 1993. [page 3]
- [48] C. Soler, K. Subr, F. Durand, N. Holzschuch, and F. Sillion, “Fourier Depth of Field,” *ACM Transactions on Graphics*, vol. 28, no. 2, pp. 18:1–18:12, May 2009. [page 3]
- [49] L. Shi, H. Hassanieh, A. Davis, D. Katabi, and F. Durand, “Light Field Reconstruction Using Sparsity in the Continuous Fourier Domain,” *ACM Transactions on Graphics*, vol. 34, no. 1, pp. 12:1–12:13, Dec 2014. [page 3]
- [50] B. Sun and R. Ramamoorthi, “Affine Double- and Triple-product Wavelet Integrals for Rendering,” *ACM Transactions on Graphics*, vol. 28, no. 2, pp. 14:1–14:17, May 2009. [page 3]
- [51] P. H. Christensen, E. J. Stollnitz, D. H. Salesin, and T. D. DeRose, “Global Illumination of Glossy Environments Using Wavelets and Importance,” *ACM Transactions on Graphics*, vol. 15, no. 1, pp. 37–71, Jan 1996. [page 3]

- [52] R. S. Overbeck, C. Donner, and R. Ramamoorthi, “Adaptive Wavelet Rendering,” *ACM Transactions on Graphics*, vol. 28, no. 5, pp. 140:1–140:12, Dec 2009. [page 3]
- [53] M. Lounsbery, T. D. DeRose, and J. Warren, “Multiresolution Analysis for Surfaces of Arbitrary Topological Type,” *ACM Transactions on Graphics*, vol. 16, no. 1, pp. 34–73, Jan 1997. [page 3]
- [54] R. Rubinstein, M. Zibulevsky, and M. Elad, “Double Sparsity: Learning Sparse Dictionaries for Sparse Signal Approximation,” *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1553–1564, March 2010. [page 3]
- [55] 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 3]
- [56] G. Easley, D. Labate, and W.-Q. Lim, “Sparse directional image representations using the discrete shearlet transform,” *Applied and Computational Harmonic Analysis*, vol. 25, no. 1, pp. 25–46, 2008. [page 3]
- [57] M. Elad, M. A. T. Figueiredo, and Y. Ma, “On the Role of Sparse and Redundant Representations in Image Processing,” *Proceedings of the IEEE*, vol. 98, no. 6, pp. 972–982, June 2010. [page 4]
- [58] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative Adversarial Nets,” in *27th International Conference on Neural Information Processing Systems - Volume 2*, ser. NIPS’14. MIT Press, 2014, pp. 2672–2680. [page 4]
- [59] T. Karras, T. Aila, S. Laine, and J. Lehtinen, “Progressive Growing of GANs for Improved Quality, Stability, and Variation,” in *Sixth International Conference on Learning Representations (ICLR 2018)*, May 2018. [page 4]
- [60] J. Yang, J. Wright, T. S. Huang, and Y. Ma, “Image Super-Resolution Via Sparse Representation,” *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, Nov 2010. [page 4]
- [61] M. Elad and M. Aharon, “Image Denoising Via Learned Dictionaries and Sparse representation,” in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2006, pp. 895–900. [page 4]
- [62] M. Fadili, J.-L. Starck, and F. Murtagh, “Inpainting and Zooming Using Sparse Representations,” *The Computer Journal*, vol. 52, no. 1, pp. 64–79, Jan 2009. [page 4]

- [63] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, “Robust Face Recognition via Sparse Representation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, Feb 2009. [pages 4, 20, and 89]
- [64] 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 4]
- [65] 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. [pages 6 and 78]
- [66] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*, 1st ed. Springer-Verlag New York, 2010. [pages 7, 20, and 72]
- [67] D. Taubman and M. Marcellin, *JPEG2000 Image Compression Fundamentals, Standards and Practice*. Springer Publishing Company, Incorporated, 2013. [page 8]
- [68] 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 8]
- [69] C. E. Shannon, “Communication in the Presence of Noise,” *Proceedings of the IRE*, vol. 37, no. 1, pp. 10–21, Jan 1949. [pages 9 and 59]
- [70] H. Nyquist, “Certain Topics in Telegraph Transmission Theory,” *Transactions of the American Institute of Electrical Engineers*, vol. 47, no. 2, pp. 617–644, April 1928. [pages 9 and 59]
- [71] *Compressed Sensing: Theory and Applications*. Cambridge University Press, 2012. [page 9]
- [72] J. . Fuchs, “On sparse representations in arbitrary redundant bases,” *IEEE Transactions on Information Theory*, vol. 50, no. 6, pp. 1341–1344, June 2004. [page 10]
- [73] 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. [pages 10 and 67]
- [74] J. Tropp, “Just Relax: Convex Programming Methods for Identifying Sparse Signals in Noise,” *Information Theory, IEEE Transactions on*, vol. 52, no. 3, pp. 1030–1051, March 2006. [pages 10 and 68]

- [75] E. J. Candes, J. Romberg, and T. Tao, “Robust Uncertainty Principles: Exact Signal Reconstruction From Highly Incomplete Frequency Information,” *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, Feb 2006. [pages 10 and 60]
- [76] E. J. Candes and T. Tao, “Decoding by Linear Programming,” *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, Dec 2005. [pages 10, 20, 60, and 68]
- [77] 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. [pages 10 and 60]
- [78] 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. [pages 10 and 69]
- [79] J. A. Tropp, “Greed is good: algorithmic results for sparse approximation,” *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2231–2242, Oct 2004. [pages 10, 67, 68, and 73]
- [80] M. A. Davenport and M. B. Wakin, “Analysis of Orthogonal Matching Pursuit Using the Restricted Isometry Property,” *IEEE Transactions on Information Theory*, vol. 56, no. 9, pp. 4395–4401, Sept 2010. [pages 10, 71, and 73]
- [81] L. Chang and J. Wu, “An Improved RIP-Based Performance Guarantee for Sparse Signal Recovery via Orthogonal Matching Pursuit,” *IEEE Transactions on Information Theory*, vol. 60, no. 9, pp. 5702–5715, Sept 2014. [page 10]
- [82] 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 10, 22, 95, 103, and 104]
- [83] 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. [pages 10 and 89]
- [84] A. Ashok and M. A. Neifeld, “Compressive Light Field Imaging,” pp. 1–12, 2010. [page 10]
- [85] 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. [pages 10 and 93]

- [86] P. Peers, D. K. Mahajan, B. Lamond, A. Ghosh, W. Matusik, R. Ramamoorthi, and P. Debevec, “Compressive Light Transport Sensing,” *ACM Transactions on Graphics*, vol. 28, no. 1, pp. 3:1–3:18, Feb 2009. [pages 10, 55, and 90]
- [87] J. P. Haldar, D. Hernando, and Z. Liang, “Compressed-Sensing MRI With Random Encoding,” *IEEE Transactions on Medical Imaging*, vol. 30, no. 4, pp. 893–903, April 2011. [page 10]
- [88] A. Majumdar, R. K. Ward, and T. Aboulnasr, “Compressed Sensing Based Real-Time Dynamic MRI Reconstruction,” *IEEE Transactions on Medical Imaging*, vol. 31, no. 12, pp. 2253–2266, Dec 2012. [pages 10 and 90]
- [89] M. A. Davenport, C. Hegde, M. F. Duarte, and R. G. Baraniuk, “Joint Manifolds for Data Fusion,” *IEEE Transactions on Image Processing*, vol. 19, no. 10, pp. 2580–2594, Oct 2010. [page 10]
- [90] J. Haupt, W. U. Bajwa, M. Rabbat, and R. Nowak, “Compressed Sensing for Networked Data,” *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 92–101, March 2008. [page 10]
- [91] D. Malioutov, M. Cetin, and A. Willsky, “A sparse signal reconstruction perspective for source localization with sensor arrays,” *Signal Processing, IEEE Transactions on*, vol. 53, no. 8, pp. 3010–3022, Aug 2005. [page 10]
- [92] M. Hyder and K. Mahata, “Direction-of-Arrival Estimation Using a Mixed $l_{2,0}$ Norm Approximation,” *Signal Processing, IEEE Transactions on*, vol. 58, no. 9, pp. 4646–4655, Sept 2010. [page 10]
- [93] K. Sun, Y. Liu, H. Meng, and X. Wang, “Adaptive Sparse Representation for Source Localization with Gain/Phase Errors,” *Sensors*, vol. 11, no. 5, pp. 4780–4793, 2011. [page 10]
- [94] D. A. Huffman, “A Method for the Construction of Minimum-Redundancy Codes,” *Proceedings of the IRE*, vol. 40, no. 9, pp. 1098–1101, Sept 1952. [page 12]
- [95] A. Mishchenko and A. Fomenko, *A Course of Differential Geometry and Topology*. Mir Publishers, 1988. [page 16]
- [96] M. A. O. Vasilescu and D. Terzopoulos, “TensorTextures: Multilinear Image-based Rendering,” *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 336–342, Aug 2004. [pages 17, 32, and 34]
- [97] H. Wang, Q. Wu, L. Shi, Y. Yu, and N. Ahuja, “Out-of-core Tensor Approximation of Multi-dimensional Matrices of Visual Data,” *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 527–535, Jul 2005. [pages 17, 32, 34, and 45]

- [98] Y.-T. Tsai and Z.-C. Shih, “K-clustered Tensor Approximation: A Sparse Multilinear Model for Real-time Rendering,” *ACM Transactions on Graphics*, vol. 31, no. 3, pp. 19:1–19:17, Jun 2012. [pages 17, 32, 34, and 45]
- [99] X. Sun, K. Zhou, Y. Chen, S. Lin, J. Shi, and B. Guo, “Interactive Relighting with Dynamic BRDFs,” *ACM Transactions on Graphics*, vol. 26, no. 3, Jul 2007. [page 17]
- [100] Y.-T. Tsai and Z.-C. Shih, “All-frequency precomputed radiance transfer using spherical radial basis functions and clustered tensor approximation,” *ACM Transactions on Graphics*, vol. 25, no. 3, pp. 967–976, Jul 2006. [page 17]
- [101] J. He, Q. Liu, A. G. Christodoulou, C. Ma, F. Lam, and Z. Liang, “Accelerated High-Dimensional MR Imaging With Sparse Sampling Using Low-Rank Tensors,” *IEEE Transactions on Medical Imaging*, vol. 35, no. 9, pp. 2119–2129, Sept 2016. [page 17]
- [102] C. F. Caiafa and A. Cichocki, “Multidimensional Compressed Sensing and Their Applications,” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 3, no. 6, pp. 355–380, Oct 2013. [pages 17, 22, and 112]
- [103] S. Yang, M. Wang, P. Li, L. Jin, B. Wu, and L. Jiao, “Compressive Hyperspectral Imaging via Sparse Tensor and Nonlinear Compressed Sensing,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 11, pp. 5943–5957, Nov 2015. [pages 17 and 26]
- [104] X. Ding, W. Chen, and I. J. Wassell, “Joint Sensing Matrix and Sparsifying Dictionary Optimization for Tensor Compressive Sensing,” *IEEE Transactions on Signal Processing*, vol. 65, no. 14, pp. 3632–3646, July 2017. [pages 17 and 25]
- [105] K. S. Gurumoorthy, A. Rajwade, A. Banerjee, and A. Rangarajan, “A Method for Compact Image Representation Using Sparse Matrix and Tensor Projections Onto Exemplar Orthonormal Bases,” *IEEE Transactions on Image Processing*, vol. 19, no. 2, pp. 322–334, Feb 2010. [pages 17, 30, 34, 36, and 37]
- [106] M. Rezghi, “A Novel Fast Tensor-Based Preconditioner for Image Restoration,” *IEEE Transactions on Image Processing*, vol. 26, no. 9, pp. 4499–4508, Sept 2017. [page 17]
- [107] V. de A. Coutinho, R. J. Cintra, and F. M. Bayer, “Low-Complexity Multidimensional DCT Approximations for High-Order Tensor Data Decorrelation,” *IEEE Transactions on Image Processing*, vol. 26, no. 5, pp. 2296–2310, May 2017. [page 17]

- [108] M. Lee and C. Choi, “Incremental N -Mode SVD for Large-Scale Multilinear Generative Models,” *IEEE Transactions on Image Processing*, vol. 23, no. 10, pp. 4255–4269, Oct 2014. [page 17]
- [109] Q. Wu, T. Xia, C. Chen, H. S. Lin, H. Wang, and Y. Yu, “Hierarchical Tensor Approximation of Multi-Dimensional Visual Data,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 1, pp. 186–199, Jan 2008. [page 17]
- [110] T. Schultz and H. Seidel, “Estimating Crossing Fibers: A Tensor Decomposition Approach,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1635–1642, Nov 2008. [page 17]
- [111] R. Ballester-Ripoll and R. Pajarola, “Tensor Decompositions for Integral Histogram Compression and Look-Up,” *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2018. [page 17]
- [112] T. G. Kolda and B. W. Bader, “Tensor Decompositions and Applications,” *SIAM Review*, vol. 51, no. 3, pp. 455–500, Aug 2009. [page 17]
- [113] W. Hackbusch, *Tensor Spaces and Numerical Tensor Calculus*, ser. Springer Series in Computational Mathematics. Springer Berlin Heidelberg, 2012. [page 17]
- [114] N. D. Sidiropoulos, L. D. Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, “Tensor decomposition for signal processing and machine learning,” *IEEE Transactions on Signal Processing*, vol. 65, no. 13, pp. 3551–3582, July 2017. [page 18]
- [115] C. Battaglino, G. Ballard, and T. G. Kolda, “A Practical Randomized CP Tensor Decomposition,” *SIAM Journal of Matrix Analysis Applications*, vol. 39, pp. 876–901, May 2018. [page 18]
- [116] V. Sharan and G. Valiant, “Orthogonalized ALS: A Theoretically Principled Tensor Decomposition Algorithm for Practical Use,” in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. International Convention Centre, Sydney, Australia: PMLR, Aug 2017, pp. 3095–3104. [page 18]
- [117] A. Uschmajew, “Local Convergence of the Alternating Least Squares Algorithm for Canonical Tensor Approximation,” *SIAM Journal of Matrix Analysis Applications*, vol. 33, pp. 639–652, June 2012. [page 18]

- [118] A. P. da Silva, P. Comon, and A. L. F. de Almeida, “A Finite Algorithm to Compute Rank-1 Tensor Approximations,” *IEEE Signal Processing Letters*, vol. 23, no. 7, pp. 959–963, July 2016. [page 18]
- [119] X. Wang and C. Navasca, “Low-rank Approximation of Tensors via Sparse Optimization,” *Numerical Linear Algebra with Applications*, vol. 25, no. 2, p. e2136, Dec 2017. [page 18]
- [120] A. Phan, P. Tichavský, and A. Cichocki, “Partitioned Hierarchical alternating least squares algorithm for CP tensor decomposition,” in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2017, pp. 2542–2546. [page 18]
- [121] 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. [pages 18 and 32]
- [122] 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. [pages 20 and 60]
- [123] 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 20]
- [124] 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 20]
- [125] 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. [pages 21 and 70]
- [126] 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 21 and 70]
- [127] 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. [pages 21 and 70]
- [128] D. L. Donoho, Y. Tsaig, I. Drori, and J. Starck, “Sparse Solution of Underdetermined Systems of Linear Equations by Stagewise Orthogonal Matching

- Pursuit,” *IEEE Transactions on Information Theory*, vol. 58, no. 2, pp. 1094–1121, Feb 2012. [pages 21 and 70]
- [129] T. Zhang, “Adaptive Forward-Backward Greedy Algorithm for Learning Sparse Representations,” *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4689–4708, July 2011. [pages 21 and 70]
 - [130] N. Lee, “MAP Support Detection for Greedy Sparse Signal Recovery Algorithms in Compressive Sensing,” *IEEE Transactions on Signal Processing*, vol. 64, no. 19, pp. 4987–4999, Oct 2016. [page 21]
 - [131] H. Wu and S. Wang, “Adaptive Sparsity Matching Pursuit Algorithm for Sparse Reconstruction,” *IEEE Signal Processing Letters*, vol. 19, no. 8, pp. 471–474, Aug 2012. [page 21]
 - [132] A. C. Gurbuz, M. Pilanci, and O. Arikan, “Expectation Maximization Based Matching Pursuit,” in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2012, pp. 3313–3316. [page 21]
 - [133] S. K. Sahoo and A. Makur, “Signal Recovery from Random Measurements via Extended Orthogonal Matching Pursuit,” *IEEE Transactions on Signal Processing*, vol. 63, no. 10, pp. 2572–2581, May 2015. [pages 21 and 70]
 - [134] A. Adler, “Covariance-Assisted Matching Pursuit,” *IEEE Signal Processing Letters*, vol. 23, no. 1, pp. 149–153, Jan 2016. [pages 21 and 70]
 - [135] Y. Chi and R. Calderbank, “Knowledge-enhanced Matching Pursuit,” in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 6576–6580. [page 21]
 - [136] N. B. Karahanoglu and H. Erdogan, “A* Orthogonal Matching Pursuit: Best-first Search for Compressed Sensing Signal Recovery,” *Digital Signal Processing*, vol. 22, no. 4, pp. 555–568, 2012. [page 21]
 - [137] M. Yaghoobi, D. Wu, and M. E. Davies, “Fast Non-Negative Orthogonal Matching Pursuit,” *IEEE Signal Processing Letters*, vol. 22, no. 9, pp. 1229–1233, Sept 2015. [page 21]
 - [138] 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 21]
 - [139] 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 21]

- [140] 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. [page 21]
- [141] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, “Least Angle Regression,” *Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004. [page 21]
- [142] S. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, “An Interior-Point Method for Large-Scale ℓ_1 -Regularized Least Squares,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 606–617, Dec 2007. [page 21]
- [143] Z. Yang, C. Zhang, J. Deng, and W. Lu, “Orthonormal Expansion ℓ_1 -Minimization Algorithms for Compressed Sensing,” *IEEE Transactions on Signal Processing*, vol. 59, no. 12, pp. 6285–6290, Dec 2011. [page 21]
- [144] 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 21]
- [145] J. Yang and Y. Zhang, “Alternating Direction Algorithms for ℓ_1 -Problems in Compressive Sensing,” *SIAM Journal on Scientific Computing*, vol. 33, no. 1, pp. 250–278, Feb 2011. [page 21]
- [146] H. Liu, B. Song, H. Qin, and Z. Qiu, “An adaptive-admm algorithm with support and signal value detection for compressed sensing,” *IEEE Signal Processing Letters*, vol. 20, no. 4, pp. 315–318, April 2013. [page 21]
- [147] J. F. C. Mota, J. M. F. Xavier, P. M. Q. Aguiar, and M. Puschel, “Distributed Basis Pursuit,” *IEEE Transactions on Signal Processing*, vol. 60, no. 4, pp. 1942–1956, April 2012. [page 21]
- [148] 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 21]
- [149] 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 22]
- [150] 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 22]

- [151] 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 22]
- [152] T. Blumensath and M. E. Davies, “Iterative Thresholding for Sparse Approximations,” *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 629–654, Dec 2008. [page 22]
- [153] D. Peleg and R. Meir, “A Bilinear Formulation for Vector Sparsity Optimization,” *Signal Processing*, vol. 88, no. 2, pp. 375–389, 2008. [page 22]
- [154] I. F. Gorodnitsky and B. D. Rao, “Sparse Signal Reconstruction from Limited Data Using FOCUSS: A Re-weighted Minimum Norm Algorithm,” *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 600–616, March 1997. [page 22]
- [155] 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 22]
- [156] C. F. Caiafa and A. Cichocki, “Computing Sparse Representations of Multidimensional Signals Using Kronecker Bases,” *Neural Computation*, vol. 25, no. 1, pp. 186–220, Jan 2013. [page 22]
- [157] 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. [pages 22 and 66]
- [158] X. Ding, W. Chen, and I. Wassell, “Nonconvex Compressive Sensing Reconstruction for Tensor Using Structures in Modes,” in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 4658–4662. [page 22]
- [159] N. D. Sidiropoulos and A. Kyrillidis, “Multi-Way Compressed Sensing for Sparse Low-Rank Tensors,” *IEEE Signal Processing Letters*, vol. 19, no. 11, pp. 757–760, Nov 2012. [page 22]
- [160] S. Friedland, Q. Li, and D. Schonfeld, “Compressive Sensing of Sparse Tensors,” *IEEE Transactions on Image Processing*, vol. 23, no. 10, pp. 4438–4447, Oct 2014. [pages 22 and 104]
- [161] W. Dai, Y. Li, J. Zou, H. Xiong, and Y. F. Zheng, “Fully Decomposable Compressive Sampling With Joint Optimization for Multidimensional Sparse Representation,” *IEEE Transactions on Signal Processing*, vol. 66, no. 3, pp. 603–616, Feb 2018. [page 22]

- [162] 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 22, 34, and 43]
- [163] R. Rubinstein, M. Zibulevsky, and M. Elad, “Efficient Implementation of the K-SVD Algorithm Using Batch Orthogonal Matching Pursuit,” Computer Science Department, Technion–Israel Institute of Technology, Tech. Rep., Aug 2008. [page 23]
- [164] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, “Online Dictionary Learning for Sparse Coding,” in *Proceedings of the 26th Annual International Conference on Machine Learning*, ser. ICML ’09, 2009, pp. 689–696. [pages 24, 95, and 108]
- [165] S. Mukherjee, R. Basu, and C. S. Seelamantula, “ ℓ_1 -K-SVD: A robust dictionary learning algorithm with simultaneous update,” *Signal Processing*, vol. 123, pp. 42–52, June 2016. [pages 24 and 43]
- [166] M. Yaghoobi, T. Blumensath, and M. E. Davies, “Dictionary Learning for Sparse Approximations With the Majorization Method,” *IEEE Transactions on Signal Processing*, vol. 57, no. 6, pp. 2178–2191, June 2009. [page 24]
- [167] D. Barchiesi and M. D. Plumbley, “Learning Incoherent Dictionaries for Sparse Approximation Using Iterative Projections and Rotations,” *IEEE Transactions on Signal Processing*, vol. 61, no. 8, pp. 2055–2065, April 2013. [page 24]
- [168] B. Wen, S. Ravishankar, and Y. Bresler, “Learning Flipping and Rotation Invariant Sparsifying Transforms,” in *2016 IEEE International Conference on Image Processing (ICIP)*, Sept 2016, pp. 3857–3861. [page 24]
- [169] A. Abdi, A. Payani, and F. Fekri, “Learning Dictionary for Efficient Signal Compression,” in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2017, pp. 3689–3693. [page 24]
- [170] C. Studer and R. G. Baraniuk, “Dictionary Learning From Sparsely Corrupted or Compressed Signals,” in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2012, pp. 3341–3344. [page 24]
- [171] Z. Sadeghipoor, M. Babaie-Zadeh, and C. Jutten, “Dictionary Learning for Sparse Decomposition: A New Criterion and Algorithm,” in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 5855–5859. [page 24]

- [172] R. Mazhar and P. D. Gader, “EK-SVD: Optimized Dictionary Design for Sparse Representations,” in *2008 19th International Conference on Pattern Recognition*, Dec 2008, pp. 1–4. [pages 24 and 43]
- [173] J. Feng, L. Song, X. Yang, and W. Zhang, “Sub Clustering K-SVD: Size Variable Dictionary Learning for Sparse Representations,” in *2009 16th IEEE International Conference on Image Processing (ICIP)*, Nov 2009, pp. 2149–2152. [page 24]
- [174] C. Rusu and B. Dumitrescu, “Stagewise K-SVD to Design Efficient Dictionaries for Sparse Representations,” *IEEE Signal Processing Letters*, vol. 19, no. 10, pp. 631–634, Oct 2012. [page 24]
- [175] M. Marsousi, K. Abhari, P. Babyn, and J. Alirezaie, “An Adaptive Approach to Learn Overcomplete Dictionaries With Efficient Numbers of Elements,” *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3272–3283, June 2014. [pages 24 and 43]
- [176] Z. Jiang, Z. Lin, and L. S. Davis, “Label Consistent K-SVD: Learning a Discriminative Dictionary for Recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2651–2664, Nov 2013. [page 24]
- [177] J. Mairal, F. Bach, and J. Ponce, “Task-Driven Dictionary Learning,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 791–804, April 2012. [page 24]
- [178] Z. You, R. Raich, X. Z. Fern, and J. Kim, “Weakly Supervised Dictionary Learning,” *IEEE Transactions on Signal Processing*, vol. 66, no. 10, pp. 2527–2541, May 2018. [page 24]
- [179] Y. Sun, Q. Liu, J. Tang, and D. Tao, “Learning Discriminative Dictionary for Group Sparse Representation,” *IEEE Transactions on Image Processing*, vol. 23, no. 9, pp. 3816–3828, Sept 2014. [page 24]
- [180] M. J. Gangeh, A. Ghodsi, and M. S. Kamel, “Kernelized Supervised Dictionary Learning,” *IEEE Transactions on Signal Processing*, vol. 61, no. 19, pp. 4753–4767, Oct 2013. [page 24]
- [181] N. Zhou, Y. Shen, J. Peng, and J. Fan, “Learning Inter-related Visual Dictionary for Object Recognition,” in *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012, pp. 3490–3497. [page 24]
- [182] Z. Zhang, F. Li, T. W. S. Chow, L. Zhang, and S. Yan, “Sparse Codes Auto-Extractor for Classification: A Joint Embedding and Dictionary Learning

- Framework for Representation,” *IEEE Transactions on Signal Processing*, vol. 64, no. 14, pp. 3790–3805, July 2016. [page 24]
- [183] 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. [page 24]
- [184] R. Yan, L. Shao, and Y. Liu, “Nonlocal Hierarchical Dictionary Learning Using Wavelets for Image Denoising,” *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4689–4698, Dec 2013. [page 24]
- [185] W. Dong, X. Li, L. Zhang, and G. Shi, “Sparsity-based Image Denoising via Dictionary Learning and Structural Clustering,” in *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2011, pp. 457–464. [page 24]
- [186] J. Liu, X. Tai, H. Huang, and Z. Huan, “A Weighted Dictionary Learning Model for Denoising Images Corrupted by Mixed Noise,” *IEEE Transactions on Image Processing*, vol. 22, no. 3, pp. 1108–1120, March 2013. [page 24]
- [187] S. Ravishankar and Y. Bresler, “MR Image Reconstruction From Highly Undersampled k-Space Data by Dictionary Learning,” *IEEE Transactions on Medical Imaging*, vol. 30, no. 5, pp. 1028–1041, May 2011. [page 24]
- [188] Q. Xu, H. Yu, X. Mou, L. Zhang, J. Hsieh, and G. Wang, “Low-Dose X-ray CT Reconstruction via Dictionary Learning,” *IEEE Transactions on Medical Imaging*, vol. 31, no. 9, pp. 1682–1697, Sept 2012. [page 24]
- [189] T. Tong, R. Wolz, P. Coupé, J. V. Hajnal, and D. Rueckert, “Segmentation of MR Images via Discriminative Dictionary Learning and Sparse Coding: Application to Hippocampus Labeling,” *NeuroImage*, vol. 76, pp. 11–23, August 2013. [page 24]
- [190] I. Tošić, S. A. Shroff, and K. Berkner, “Dictionary Learning for Incoherent Sampling with application to plenoptic imaging,” in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 1821–1825. [page 24]
- [191] H. Chang, M. K. Ng, and T. Zeng, “Reducing Artifacts in JPEG Decompression Via a Learned Dictionary,” *IEEE Transactions on Signal Processing*, vol. 62, no. 3, pp. 718–728, Feb 2014. [page 24]
- [192] M. Sadeghi, M. Babaie-Zadeh, and C. Jutten, “Learning Overcomplete Dictionaries Based on Atom-by-Atom Updating,” *IEEE Transactions on Signal Processing*, vol. 62, no. 4, pp. 883–891, Feb 2014. [pages 24 and 43]

- [193] S. Mukherjee and C. S. Seelamantula, "A Divide-and-Conquer Dictionary Learning Algorithm and its Performance Analysis," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 4712–4716. [page 24]
- [194] A. Rakotomamonjy, "Direct Optimization of the Dictionary Learning Problem," *IEEE Transactions on Signal Processing*, vol. 61, no. 22, pp. 5495–5506, Nov 2013. [page 24]
- [195] G. Peng and W. Hwang, "A Proximal Method for Dictionary Updating in Sparse Representations," *IEEE Transactions on Signal Processing*, vol. 63, no. 15, pp. 3946–3958, Aug 2015. [page 24]
- [196] J. Liang, M. Zhang, X. Zeng, and G. Yu, "Distributed dictionary learning for sparse representation in sensor networks," *IEEE Transactions on Image Processing*, vol. 23, no. 6, pp. 2528–2541, June 2014. [page 24]
- [197] J. Sulam, V. Pappyan, Y. Romano, and M. Elad, "Multilayer Convolutional Sparse Modeling: Pursuit and Dictionary Learning," *IEEE Transactions on Signal Processing*, vol. 66, no. 15, pp. 4090–4104, Aug 2018. [page 25]
- [198] F. Roemer, G. D. Galdo, and M. Haardt, "Tensor-based Algorithms for Learning Multidimensional Separable Dictionaries," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 3963–3967. [page 25]
- [199] K. Engan, S. O. Aase, and J. H. Husoy, "Method of Optimal Directions for Frame Design," in *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99*, vol. 5, March 1999, pp. 2443–2446. [page 25]
- [200] S. Hawe, M. Seibert, and M. Kleinstenber, "Separable Dictionary Learning," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, June 2013, pp. 438–445. [page 25]
- [201] P. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2009. [page 25]
- [202] M. E. Gehm, R. John, D. J. Brady, R. M. Willett, and T. J. Schulz, "Single-shot Compressive Spectral Imaging with a Dual-disperser Architecture," *Optics Express*, vol. 15, no. 21, pp. 14013–14027, Oct 2007. [page 26]
- [203] D. Kittle, K. Choi, A. Wagadarikar, and D. J. Brady, "Multiframe Image Estimation for Coded Aperture Snapshot Spectral Imagers," *Applied Optics*, vol. 49, no. 36, pp. 6824–6833, Dec 2010. [pages 26 and 95]

- [204] H. Arguello and G. R. Arce, “Colored Coded Aperture Design by Concentration of Measure in Compressive Spectral Imaging,” *IEEE Transactions on Image Processing*, vol. 23, no. 4, pp. 1896–1908, April 2014. [page 26]
- [205] D. N. Wood, D. I. Azuma, K. Aldinger, B. Curless, T. Duchamp, D. H. Salesin, and W. Stuetzle, “Surface Light Fields for 3D Photography,” in *Proceedings of SIGGRAPH 2000*. ACM Press/Addison-Wesley Publishing Co., 2000, pp. 287–296. [page 27]
- [206] A. Isaksen, L. McMillan, and S. J. Gortler, “Dynamically Reparameterized Light Fields,” in *Proceedings of SIGGRAPH 2000*. ACM Press/Addison-Wesley Publishing Co., 2000, pp. 297–306. [page 27]
- [207] B. Krolla, M. Diebold, H. Collaboratory, I. Science, and D. Stricker, “Spherical Light Fields,” in *Proceedings of the British Machine Vision Conference*, T. Valstar, Michel and French, Andrew and Pridmore, Ed. BMVA Press, 2014, pp. 1–12. [page 27]
- [208] A. Davis, M. Levoy, and F. Durand, “Unstructured Light Fields,” *Computer Graphics Forum*, vol. 31, no. 2, pp. 305–314, 2012. [page 27]
- [209] Z. Xiong, L. Wang, H. Li, D. Liu, and F. Wu, “Snapshot Hyperspectral Light Field Imaging,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 6873–6881. [page 27]
- [210] Y. Zhao, T. Yue, L. Chen, H. Wang, Z. Ma, D. J. Brady, and X. Cao, “Heterogeneous Camera Array for Multispectral Light Field Imaging,” *Optics Express*, vol. 25, no. 13, pp. 14 008–14 022, Jun 2017. [page 27]
- [211] 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. [pages 29 and 41]
- [212] 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 29]
- [213] S. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang, and S. Lin, “Graph Embedding and Extensions: A General Framework for Dimensionality Reduction,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 40–51, Jan 2007. [page 29]
- [214] L. v. d. Maaten and G. Hinton, “Tensor Rank and the Ill-Posedness of the Best Low-Rank Approximation Problem,” *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, Nov 2008. [page 29]

- [215] 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 29]
- [216] S. Lesage, R. Gribonval, F. Bimbot, and L. Benaroya, “Learning Unions of Orthonormal Bases with Thresholded Singular Value Decomposition,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 2005*, vol. 5, March 2005, pp. 293–296. [pages 30, 34, and 37]
- [217] R. Horn and C. Johnson, *Matrix Analysis*. Cambridge University Press, 2012. [pages 32 and 82]
- [218] S. K. Suter, J. A. I. Guitian, F. Marton, M. Agus, A. Elsener, C. P. E. Zollikofer, M. Gopi, E. Gobbetti, and R. Pajarola, “Interactive Multiscale Tensor Reconstruction for Multiresolution Volume Visualization,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2135–2143, Dec 2011. [page 32]
- [219] J. Hou, L. P. Chau, N. Magnenat-Thalmann, and Y. He, “Scalable and compact representation for motion capture data using tensor decomposition,” *IEEE Signal Processing Letters*, vol. 21, no. 3, pp. 255–259, Mar 2014. [page 32]
- [220] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006. [page 32]
- [221] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, “MPCA: Multilinear Principal Component Analysis of Tensor Objects,” *IEEE Transactions on Neural Networks*, vol. 19, no. 1, pp. 18–39, Jan 2008. [page 32]
- [222] A. Montanari and E. Richard, “A statistical model for tensor pca,” in *Proceedings of the 27th International Conference on Neural Information Processing Systems*, ser. NIPS’14. Cambridge, MA, USA: MIT Press, 2014, pp. 2897–2905. [page 32]
- [223] 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 33]
- [224] 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 33]
- [225] 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 33]

- [226] D. Mahajan, I. K. Shlizerman, R. Ramamoorthi, and P. Belhumeur, “A Theory of Locally Low Dimensional Light Transport,” *ACM Transactions on Graphics*, vol. 26, no. 3, Jul 2007. [page 33]
- [227] W. Dong, L. Zhang, G. Shi, and X. Li, “Nonlocally Centralized Sparse Representation for Image Restoration,” *IEEE Transactions on Image Processing*, vol. 22, no. 4, pp. 1620–1630, April 2013. [page 33]
- [228] P.-P. Sloan, J. Hall, J. Hart, and J. Snyder, “Clustered Principal Components for Precomputed Radiance Transfer,” *ACM Transactions on Graphics*, vol. 22, no. 3, pp. 382–391, Jul 2003. [pages 33, 39, and 45]
- [229] K. Schröder, R. Klein, and A. Zinke, “Non-Local Image Reconstruction for Efficient Computation of Synthetic Bidirectional Texture Functions,” *Computer Graphics Forum*, vol. 32, no. 8, pp. 61–71, June 2013. [page 33]
- [230] 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 34]
- [231] A. Rangarajan, A. Yuille, and E. Mjolsness, “Convergence Properties of the Softassign Quadratic Assignment Algorithm,” *Neural Computation*, vol. 11, no. 6, pp. 1455–1474, Aug 1999. [page 36]
- [232] J. Kosowsky and A. Yuille, “The Invisible Hand Algorithm: Solving the Assignment Problem with Statistical Physics,” *Neural Networks*, vol. 7, no. 3, pp. 477–490, 1994. [page 36]
- [233] K. B. Petersen and M. S. Pedersen, “The matrix cookbook,” Nov 2012, version 20121115. [page 37]
- [234] R. Wang, R. Wang, K. Zhou, M. Pan, and H. Bao, “An Efficient GPU-based Approach for Interactive Global Illumination,” *ACM Transactions Graphics*, vol. 28, no. 3, pp. 91:1–91:8, Jul 2009. [page 39]
- [235] A. Bock, E. Sundén, B. Liu, B. Wünsche, and T. Ropinski, “Coherency-Based Curve Compression for High-Order Finite Element Model Visualization,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2315–2324, Dec 2012. [page 39]
- [236] Z. Zheng, N. Ahmed, and K. Mueller, “iView: A Feature Clustering Framework for Suggesting Informative Views in Volume Visualization,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 1959–1968, Dec 2011. [page 39]

- [237] C. W. Chen, J. Luo, and K. J. Parker, “Image Segmentation via Adaptive K-mean Clustering and Knowledge-based Morphological Operations with Biomedical Applications,” *IEEE Transactions on Image Processing*, vol. 7, no. 12, pp. 1673–1683, Dec 1998. [page 39]
- [238] A. Y. Ng, M. I. Jordan, and Y. Weiss, “On Spectral Clustering: Analysis and an Algorithm,” in *Proceedings of the 14th International Conference on Neural Information Processing Systems*, ser. NIPS’01. MIT Press, Dec 2001, pp. 849–856. [page 39]
- [239] E. Elhamifar and R. Vidal, “Sparse Subspace Clustering,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 2790–2797. [page 39]
- [240] H. Xu, C. Caramanis, and S. Mannor, “Outlier-Robust PCA: The High-Dimensional Case,” *IEEE Transactions on Information Theory*, vol. 59, no. 1, pp. 546–572, Jan 2013. [page 41]
- [241] 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 41]
- [242] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma, “Robust Principal Component Analysis: Exact Recovery of Corrupted Low-Rank Matrices via Convex Optimization,” in *Advances in Neural Information Processing Systems (NIPS 2009)*, Y. Bengio, D. Schuurmans, J. D. Lafferty, C. K. I. Williams, and A. Culotta, Eds. Curran Associates, Inc., 2009, pp. 2080–2088. [page 41]
- [243] Y. Fang, L. Chen, J. Wu, and B. Huang, “GPU Implementation of Orthogonal Matching Pursuit for Compressive Sensing,” in *2011 IEEE 17th International Conference on Parallel and Distributed Systems*, Dec 2011, pp. 1044–1047. [pages 43 and 72]
- [244] F. Huang, J. Tao, Y. Xiang, P. Liu, L. Dong, and L. Wang, “Parallel Compressive Sampling Matching Pursuit Algorithm for Compressed Sensing Signal Reconstruction with OpenCL,” *Journal of Systems Architecture*, vol. 72, no. C, pp. 51–60, Jan 2017. [page 43]
- [245] J. D. Blanchard and J. Tanner, “GPU Accelerated Greedy Algorithms for Compressed Sensing,” *Mathematical Programming Computation*, vol. 5, no. 3, pp. 267–304, Sept 2013. [page 43]
- [246] C.-L. Chang, X. Zhu, P. Ramanathan, and B. Girod, “Light field compression using disparity-compensated lifting and shape adaptation,” *IEEE*

- Transactions on Image Processing*, vol. 15, no. 4, pp. 793–806, April 2006. [page 45]
- [247] N. Fout and K. Ma, “An Adaptive Prediction-Based Approach to Lossless Compression of Floating-Point Volume Data,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2295–2304, Dec 2012. [page 45]
- [248] G. Müller, J. Meseth, and R. Klein, “Compression and Real-Time Rendering of Measured BTFs Using Local PCA,” in *The 8th International Fall Workshop Vision, Modeling and Visualisation 2003*, T. Ertl, B. Girod, G. Greiner, H. Niemann, H.-P. Seidel, E. Steinbach, and R. Westermann, Eds. Akademische Verlagsgesellschaft Aka GmbH, Berlin, Nov 2003, pp. 271–280. [page 45]
- [249] A. Appel, “Some Techniques for Shading Machine Renderings of Solids,” in *Proceedings of the April 30–May 2, 1968, Spring Joint Computer Conference*. ACM, 1968, pp. 37–45. [pages 46 and 98]
- [250] J. T. Kajiya, “The Rendering Equation,” in *Proceedings of SIGGRAPH 1986*. ACM, 1986, pp. 143–150. [pages 46 and 98]
- [251] S. Lee, E. Eisemann, and H.-P. Seidel, “Real-time Lens Blur Effects and Focus Control,” *ACM Transactions Graphics*, vol. 29, no. 4, pp. 65:1–65:7, Jul 2010. [page 47]
- [252] M. Hullin, E. Eisemann, H.-P. Seidel, and S. Lee, “Physically-based Real-time Lens Flare Rendering,” *ACM Transactions Graphics*, vol. 30, no. 4, pp. 108:1–108:10, Jul 2011. [page 47]
- [253] L. Belcour, “Efficient Rendering of Layered Materials Using an Atomic Decomposition with Statistical Operators,” *ACM Transactions Graphics*, vol. 37, no. 4, pp. 73:1–73:15, Jul 2018. [page 47]
- [254] A. Toisoul and A. Ghosh, “Practical Acquisition and Rendering of Diffraction Effects in Surface Reflectance,” *ACM Transactions Graphics*, vol. 36, no. 4, Jul 2017. [page 47]
- [255] I. Sadeghi, A. Munoz, P. Laven, W. Jarosz, F. Seron, D. Gutierrez, and H. W. Jensen, “Physically-based Simulation of Rainbows,” *ACM Transactions Graphics*, vol. 31, no. 1, pp. 3:1–3:12, Feb 2012. [page 47]
- [256] K. Xu, L.-Q. Ma, B. Ren, R. Wang, and S.-M. Hu, “Interactive Hair Rendering and Appearance Editing Under Environment Lighting,” *ACM Transactions Graphics*, vol. 30, no. 6, pp. 173:1–173:10, Dec 2011. [page 47]

- [257] J. M. Singh and P. J. Narayanan, “Real-Time Ray Tracing of Implicit Surfaces on the GPU,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. 2, pp. 261–272, March 2010. [page 47]
- [258] I. Wald, H. Friedrich, G. Marmitt, P. Slusallek, and H. P. Seidel, “Faster isosurface ray tracing using implicit KD-trees,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 11, no. 5, pp. 562–572, Sept 2005. [page 47]
- [259] A. Silvennoinen and J. Lehtinen, “Real-time Global Illumination by Precomputed Local Reconstruction from Sparse Radiance Probes,” *ACM Transactions Graphics*, vol. 36, no. 6, pp. 230:1–230:13, Nov. 2017. [page 47]
- [260] P. Ren, J. Wang, M. Gong, S. Lin, X. Tong, and B. Guo, “Global Illumination with Radiance Regression Functions,” *ACM Transactions Graphics*, vol. 32, no. 4, pp. 130:1–130:12, Jul 2013. [page 47]
- [261] P. Shirley and K. Chiu, “A Low Distortion Map Between Disk and Square,” *Journal of Graphics Tools*, vol. 2, no. 3, pp. 45–52, 1997. [page 48]
- [262] N. Sabater, G. Boisson, B. Vandame, P. Kerbirou, 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 53, 54, 103, and 104]
- [263] M. H. Kamal, B. Heshmat, R. Raskar, P. Vanderghenst, and G. Wetzstein, “Tensor low-rank and sparse light field photography,” *Computer Vision and Image Understanding*, vol. 145, pp. 172–181, 2016, light Field for Computer Vision. [page 52]
- [264] J.-D. Durou, M. Falcone, and M. Sagona, “Numerical Methods for Shape-from-shading: A New Survey with Benchmarks,” *Computer Vision and Image Understanding*, vol. 109, no. 1, pp. 22–43, Jan 2008. [page 55]
- [265] Lucas Digital Ltd. LLC., *Technical Introduction to OpenEXR*, Nov 2006. [page 55]
- [266] I. Ihm, S. Park, and R. K. Lee, “Rendering of Spherical Light Fields,” in *Proceedings The Fifth Pacific Conference on Computer Graphics and Applications*, Oct 1997, pp. 59–68. [page 57]
- [267] F. Marvasti, Ed., *Nonuniform Sampling: Theory and Practice*, ser. Information Technology: Transmission, Processing and Storage. Springer US, 2001. [page 59]

- [268] E. Margolis and Y. C. Eldar, “Nonuniform Sampling of Periodic Bandlimited Signals,” *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 2728–2745, July 2008. [page 59]
- [269] H. J. Landau, “Necessary Density Conditions for Sampling and Interpolation of Certain Entire Functions,” *Acta Mathematica*, vol. 117, pp. 37–52, 1967. [page 59]
- [270] K. Yao and J. Thomas, “On some stability and interpolatory properties of nonuniform sampling expansions,” *IEEE Transactions on Circuit Theory*, vol. 14, no. 4, pp. 404–408, December 1967. [page 59]
- [271] Y.-P. Lin and P. P. Vaidyanathan, “Periodically Nonuniform Sampling of Bandpass Signals,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 45, no. 3, pp. 340–351, March 1998. [page 59]
- [272] Z. Song, B. Liu, Y. Pang, C. Hou, and X. Li, “An Improved Nyquist–Shannon Irregular Sampling Theorem From Local Averages,” *IEEE Transactions on Information Theory*, vol. 58, no. 9, pp. 6093–6100, Sept 2012. [page 59]
- [273] A. J. Jerri, “The Shannon Sampling Theorem – Its Various Extensions and Applications: A Tutorial Review,” *Proceedings of the IEEE*, vol. 65, no. 11, pp. 1565–1596, Nov 1977. [page 59]
- [274] H. Stark, *Polar, Spiral, and Generalized Sampling and Interpolation*. New York, NY: Springer US, 1993, pp. 185–218. [page 60]
- [275] L. A. Shepp and B. F. Logan, “The Fourier Reconstruction of a Head Section,” *IEEE Transactions on Nuclear Science*, vol. 21, no. 3, pp. 21–43, June 1974. [page 60]
- [276] R. R. Naidu and C. R. Murthy, “Construction of Binary Sensing Matrices Using Extremal Set Theory,” *IEEE Signal Processing Letters*, vol. 24, no. 2, pp. 211–215, Feb 2017. [page 64]
- [277] A. Amini and F. Marvasti, “Deterministic Construction of Binary, Bipolar, and Ternary Compressed Sensing Matrices,” *IEEE Transactions on Information Theory*, vol. 57, no. 4, pp. 2360–2370, April 2011. [page 64]
- [278] S. Li, F. Gao, G. Ge, and S. Zhang, “Deterministic Construction of Compressed Sensing Matrices via Algebraic Curves,” *IEEE Transactions on Information Theory*, vol. 58, no. 8, pp. 5035–5041, Aug 2012. [page 64]
- [279] H. Monajemi, S. Jafarpour, M. Gavish, and D. L. Donoho, “Deterministic Matrices Matching the Compressed Sensing Phase Transitions of Gaussian

- Random Matrices,” *Proceedings of the National Academy of Sciences*, vol. 110, no. 4, pp. 1181–1186, Nov 2013. [page 64]
- [280] S. Li and G. Ge, “Deterministic Sensing Matrices Arising From Near Orthogonal Systems,” *IEEE Transactions on Information Theory*, vol. 60, no. 4, pp. 2291–2302, April 2014. [page 64]
- [281] R. A. DeVore, “Deterministic Constructions of Compressed Sensing Matrices,” *Journal of Complexity*, vol. 23, no. 4, pp. 918–925, Aug 2007. [page 64]
- [282] R. R. Naidu, P. Jampana, and C. S. Sastry, “Deterministic Compressed Sensing Matrices: Construction via Euler Squares and Applications,” *IEEE Transactions on Signal Processing*, vol. 64, no. 14, pp. 3566–3575, July 2016. [page 64]
- [283] P. Sasmal, R. R. Naidu, C. S. Sastry, and P. Jampana, “Composition of Binary Compressed Sensing Matrices,” *IEEE Signal Processing Letters*, vol. 23, no. 8, pp. 1096–1100, Aug 2016. [page 64]
- [284] N. Cleju, “Optimized projections for compressed sensing via rank-constrained nearest correlation matrix,” *Applied and Computational Harmonic Analysis*, vol. 36, no. 3, pp. 495–507, May 2014. [page 64]
- [285] H. Bai, S. Li, and X. He, “Sensing Matrix Optimization Based on Equiangular Tight Frames With Consideration of Sparse Representation Error,” *IEEE Transactions on Multimedia*, vol. 18, no. 10, pp. 2040–2053, Oct 2016. [page 64]
- [286] Q. Zhang, Y. Fu, H. Li, and R. Rong, “Optimized Projection Matrix for Compressed Sensing,” *Circuits, Systems, and Signal Processing*, vol. 33, no. 5, pp. 1627–1636, May 2014. [page 64]
- [287] J. M. Duarte-Carvajalino and G. Sapiro, “Learning to Sense Sparse Signals: Simultaneous Sensing Matrix and Sparsifying Dictionary Optimization,” *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1395–1408, July 2009. [page 64]
- [288] K. Li, L. Gan, and C. Ling, “Convolutional Compressed Sensing Using Deterministic Sequences,” *IEEE Transactions on Signal Processing*, vol. 61, no. 3, pp. 740–752, Feb 2013. [page 64]
- [289] D. Valsesia and E. Magli, “Compressive Signal Processing with Circulant Sensing Matrices,” in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 1015–1019. [page 64]
- [290] B. M. Sanandaji, T. L. Vincent, and M. B. Wakin, “Concentration of Measure Inequalities for Compressive Toeplitz Matrices with Applications to Detection

- and System Identification,” in *49th IEEE Conference on Decision and Control (CDC)*, Dec 2010, pp. 2922–2929. [page 64]
- [291] H. Rauhut, J. Romberg, and J. A. Tropp, “Restricted Isometries for Partial Random Circulant Matrices,” *Applied and Computational Harmonic Analysis*, vol. 32, no. 2, pp. 242–254, 2012. [page 64]
- [292] W. Yin, S. Morgan, J. Yang, and Y. Zhang, “Practical Compressive Sensing with Toeplitz and Circulant Matrices,” *Proceedings of SPIE*, vol. 7744, p. 110, 2010. [page 64]
- [293] M. Pinsky, *Introduction to Fourier Analysis and Wavelets*, ser. Graduate studies in mathematics. American Mathematical Society, 2008. [page 65]
- [294] D. Donoho and P. Stark, “Uncertainty Principles and Signal Recovery,” *SIAM Journal on Applied Mathematics*, vol. 49, no. 3, pp. 906–931, July 1989. [page 65]
- [295] D. L. Donoho and X. Huo, “Uncertainty principles and ideal atomic decomposition,” *IEEE Transactions on Information Theory*, vol. 47, no. 7, pp. 2845–2862, Nov 2001. [pages 65 and 66]
- [296] M. Elad and A. M. Bruckstein, “A Generalized Uncertainty Principle and Sparse Representation in Pairs of Bases,” *IEEE Transactions on Information Theory*, vol. 48, no. 9, pp. 2558–2567, Sept 2002. [page 66]
- [297] S. Gleichman and Y. C. Eldar, “Blind Compressed Sensing,” *IEEE Transactions on Information Theory*, vol. 57, no. 10, pp. 6958–6975, Oct 2011. [pages 67, 81, and 82]
- [298] 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 69]
- [299] T. T. Cai, L. Wang, and G. Xu, “New Bounds for Restricted Isometry Constants,” *IEEE Transactions on Information Theory*, vol. 56, no. 9, pp. 4388–4394, Sept 2010. [page 69]
- [300] T. T. Cai, L. Wang, and G. Xu, “Shifting Inequality and Recovery of Sparse Signals,” *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1300–1308, March 2010. [page 69]
- [301] S. Foucart, “A Note on Guaranteed Sparse Recovery via ℓ_1 -Minimization,” *Applied and Computational Harmonic Analysis*, vol. 29, no. 1, pp. 97–103, July 2010. [page 69]

- [302] Q. Mo and S. Li, “New Bounds on the Restricted Isometry Constant δ_{2k} ,” *Applied and Computational Harmonic Analysis*, vol. 31, no. 3, pp. 460–468, Nov 2011. [page 69]
- [303] 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. [pages 70 and 90]
- [304] Y. Arjoune, N. Kaabouch, H. E. Ghazi, and A. Tamtaoui, “Compressive Sensing: Performance Comparison of Sparse Recovery Algorithms,” in *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*, Jan 2017, pp. 1–7. [page 72]
- [305] F. Marvasti, A. Amini, F. Haddadi, M. Soltanolkotabi, B. H. Khalaj, A. Aldroubi, S. Sanei, and J. Chambers, “A Unified Approach to Sparse Signal Processing,” *EURASIP Journal on Advances in Signal Processing*, vol. 2012, no. 1, p. 44, Feb 2012. [page 72]
- [306] L. Bai, P. Maechler, M. Muehlberghuber, and H. Kaeslin, “High-speed Compressed Sensing Reconstruction on FPGA Using OMP and AMP,” in *2012 19th IEEE International Conference on Electronics, Circuits, and Systems (ICECS 2012)*, Dec 2012, pp. 53–56. [page 72]
- [307] H. Rabah, A. Amira, B. K. Mohanty, S. Almaadeed, and P. K. Meher, “FPGA Implementation of Orthogonal Matching Pursuit for Compressive Sensing Reconstruction,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 10, pp. 2209–2220, Oct 2015. [page 72]
- [308] A. Kulkarni and T. Mohsenin, “Accelerating Compressive Sensing Reconstruction OMP Algorithm with CPU, GPU, FPGA and Domain Specific Many-core,” in *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2015, pp. 970–973. [page 72]
- [309] C. Herzet, C. Soussen, J. Idier, and R. Gribonval, “Exact Recovery Conditions for Sparse Representations With Partial Support Information,” *IEEE Transactions on Information Theory*, vol. 59, no. 11, pp. 7509–7524, Nov 2013. [page 73]
- [310] C. Soussen, R. Gribonval, J. Idier, and C. Herzet, “Joint K-Step Analysis of Orthogonal Matching Pursuit and Orthogonal Least Squares,” *IEEE Transactions on Information Theory*, vol. 59, no. 5, pp. 3158–3174, May 2013. [page 73]

- [311] Q. Mo and Y. Shen, “A Remark on the Restricted Isometry Property in Orthogonal Matching Pursuit,” *IEEE Transactions on Information Theory*, vol. 58, no. 6, pp. 3654–3656, June 2012. [page 73]
- [312] J. Wang and B. Shim, “On the Recovery Limit of Sparse Signals Using Orthogonal Matching Pursuit,” *IEEE Transactions on Signal Processing*, vol. 60, no. 9, pp. 4973–4976, Sept 2012. [page 73]
- [313] R. Wu, W. Huang, and D. Chen, “The Exact Support Recovery of Sparse Signals With Noise via Orthogonal Matching Pursuit,” *IEEE Signal Processing Letters*, vol. 20, no. 4, pp. 403–406, April 2013. [page 73]
- [314] J. Wen, Z. Zhou, J. Wang, X. Tang, and Q. Mo, “A Sharp Condition for Exact Support Recovery With Orthogonal Matching Pursuit,” *IEEE Transactions on Signal Processing*, vol. 65, no. 6, pp. 1370–1382, March 2017. [page 73]
- [315] J. Wang and B. Shim, “Exact Recovery of Sparse Signals Using Orthogonal Matching Pursuit: How Many Iterations Do We Need?” *IEEE Transactions on Signal Processing*, vol. 64, no. 16, pp. 4194–4202, Aug 2016. [page 73]
- [316] J. Ding, L. Chen, and Y. Gu, “Perturbation Analysis of Orthogonal Matching Pursuit,” *IEEE Transactions on Signal Processing*, vol. 61, no. 2, pp. 398–410, Jan 2013. [page 73]
- [317] A. M. Tillmann and M. E. Pfetsch, “The computational complexity of the restricted isometry property, the nullspace property, and related concepts in compressed sensing,” *IEEE Transactions on Information Theory*, vol. 60, no. 2, pp. 1248–1259, Feb 2014. [page 74]
- [318] G. Bennett, “Probability Inequalities for the Sum of Independent Random Variables,” *Journal of the American Statistical Association*, vol. 57, no. 297, pp. 33–45, 1962. [page 76]
- [319] S. Boucheron, G. Lugosi, and P. Massart, *Concentration Inequalities: A Nonasymptotic Theory of Independence*. OUP Oxford, 2013. [page 76]
- [320] J. A. Tropp, “User-Friendly Tail Bounds for Sums of Random Matrices,” *Foundations of Computational Mathematics*, vol. 12, no. 4, pp. 389–434, Aug 2012. [page 83]
- [321] C. Luo, F. Wu, J. Sun, and C. W. Chen, “Compressive Data Gathering for Large-scale Wireless Sensor Networks,” in *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom ’09. ACM, 2009, pp. 145–156. [page 89]

- [322] Z. Tian and G. B. Giannakis, "Compressed Sensing for Wideband Cognitive Radios," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing*, ser. ICASSP '07, April 2007, pp. IV-1357-IV-1360. [page 89]
- [323] P. Nagesh and B. Li, "A Compressive Sensing Approach for Expression-Invariant Face Recognition," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 1518-1525. [page 89]
- [324] J. F. Gemmeke, H. V. Hamme, B. Cranen, and L. Boves, "Compressive Sensing for Missing Data Imputation in Noise Robust Speech Recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 272-287, April 2010. [page 89]
- [325] A. Akl, C. Feng, and S. Valaee, "A Novel Accelerometer-Based Gesture Recognition System," *IEEE Transactions on Signal Processing*, vol. 59, no. 12, pp. 6197-6205, Dec 2011. [page 89]
- [326] A. Amir and O. Zuk, "Bacterial Community Reconstruction Using Compressed Sensing," *Journal of computational biology*, vol. 18, no. 11, pp. 1723-1741, Nov 2011. [page 89]
- [327] W. TANG, H. CAO, J. DUAN, and Y.-P. WANG, "A Compressed Sensing Based Approach for subtyping of Leukemia from Gene Expression Data," *Journal of Bioinformatics and Computational Biology*, vol. 09, no. 05, pp. 631-645, 2011. [page 89]
- [328] V. M. Patel, G. R. Easley, J. D. M. Healy, and R. Chellappa, "Compressed Synthetic Aperture Radar," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 244-254, April 2010. [page 89]
- [329] A. C. Gurbuz, J. H. McClellan, and W. R. S. Jr., "Compressive Sensing for Subsurface Imaging Using Ground Penetrating Radar," *Signal Processing*, vol. 89, no. 10, pp. 1959-1972, Oct 2009. [page 89]
- [330] D. Shamsi, P. Boufounos, and F. Koushanfar, "Noninvasive Leakage Power Tomography of Integrated Circuits by Compressive Sensing," in *Proceeding of the 13th international symposium on Low power electronics and design (ISLPED '08)*, Aug 2008, pp. 341-346. [page 89]
- [331] D. Gangopadhyay, E. G. Allstot, A. M. R. Dixon, K. Natarajan, S. Gupta, and D. J. Allstot, "Compressed Sensing Analog Front-End for Bio-Sensor Applications," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 2, pp. 426-438, Feb 2014. [page 89]

- [332] C. Liao, J. Tao, X. Zeng, Y. Su, D. Zhou, and X. Li, “Efficient Spatial Variation Modeling of Nanoscale Integrated Circuits Via Hidden Markov Tree,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 6, pp. 971–984, June 2016. [page 89]
- [333] T. Xu and W. Wang, “A Compressed Sensing Approach for Underdetermined Blind Audio Source Separation with Sparse Representation,” in *2009 IEEE/SP 15th Workshop on Statistical Signal Processing*, Aug 2009, pp. 493–496. [page 89]
- [334] G. Bao, Z. Ye, X. Xu, and Y. Zhou, “A Compressed Sensing Approach to Blind Separation of Speech Mixture Based on a Two-Layer Sparsity Model,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 5, pp. 899–906, May 2013. [page 89]
- [335] F. M. aes, F. M. Araújo, M. V. Correia, M. Abolbashari, and F. Farahi, “Active Illumination Single-pixel Camera Based on Compressive Sensing,” *Applied Optics*, vol. 50, no. 4, pp. 405–414, Feb 2011. [page 89]
- [336] Z. Li, J. Suo, X. Hu, C. Deng, J. Fan, and Q. Dai, “Efficient Single-pixel Multi-spectral Imaging via Non-mechanical Spatio-spectral Modulation,” *Scientific Reports*, vol. 7, no. 41435, Jan 2017. [page 90]
- [337] H. Garcia, C. V. Correa, and H. Arguello, “Multi-Resolution Compressive Spectral Imaging Reconstruction From Single Pixel Measurements,” *IEEE Transactions on Image Processing*, vol. 27, no. 12, pp. 6174–6184, Dec 2018. [page 90]
- [338] M. P. Edgar, G. M. Gibson, R. W. Bowman, B. Sun, N. Radwell, K. J. Mitchell, S. S. Welsh, and M. J. Padgett1, “Simultaneous Real-time Visible and Infrared Video with Single-pixel Detectors,” *Scientific Reports*, vol. 5, no. 10669, May 2015. [page 90]
- [339] 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, Nov 2014. [page 90]
- [340] W. L. Chan, K. Charan, D. Takhar, K. F. Kelly, R. G. Baraniuk, and D. M. Mittleman, “A Single-pixel Terahertz Imaging System Based on Compressed Sensing,” *Applied Physics Letters*, vol. 93, no. 12, p. 121105, Sep 2008. [page 90]
- [341] Z. Fang, N. Van Le, M. Choy, and J. H. Lee, “High Spatial Resolution Compressed Sensing (HSPARSE) Functional MRI,” *Magnetic Resonance in Medicine*, vol. 76, no. 2, pp. 440–455, Oct 2015. [page 90]

- [342] W. Lu, T. Li, I. C. Atkinson, and N. Vaswani, “Modified-CS-residual for Recursive Reconstruction of Highly Undersampled Functional MRI Sequences,” in *2011 18th IEEE International Conference on Image Processing*, Sept 2011, pp. 2689–2692. [page 90]
- [343] X. Zong, J. Lee, A. J. Poplawsky, S.-G. Kim, and J. C. Ye, “Compressed Sensing fMRI Using Gradient-recalled Echo and EPI Sequences,” *NeuroImage*, vol. 92, pp. 312–321, May 2014. [page 90]
- [344] D. J. Holland, C. Liu, X. Song, E. L. Mazerolle, M. T. Stevens, A. J. Sederman, L. F. Gladden, R. C. N. D’Arcy, C. V. Bowen, and S. D. Beyea, “Compressed Sensing Reconstruction Improves Sensitivity of Variable Density Spiral fMRI,” *Magnetic Resonance in Medicine*, vol. 70, no. 6, pp. 1634–1643, Feb 2013. [page 90]
- [345] U. Gamper, P. Boesiger, and S. Kozerke, “Compressed Sensing in Dynamic MRI,” *Magnetic Resonance in Medicine*, vol. 59, no. 2, pp. 365–373, Jan 2008. [page 90]
- [346] C. Qiu, W. Lu, and N. Vaswani, “Real-time Dynamic MR Image Reconstruction Using Kalman Filtered Compressed Sensing,” in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, April 2009, pp. 393–396. [page 90]
- [347] R. Otazo, E. Candès, and D. K. Sodickson, “Low-rank Plus Sparse Matrix Decomposition for Accelerated Dynamic MRI with Separation of Background and Dynamic Components,” *Magnetic Resonance in Medicine*, vol. 73, no. 3, pp. 1125–1136, April 2014. [page 90]
- [348] L. Feng, R. Grimm, K. T. Block, H. Chandarana, S. Kim, J. Xu, L. Axel, D. K. Sodickson, and R. Otazo, “Golden-angle Radial Sparse Parallel MRI: Combination of Compressed Sensing, Parallel Imaging, and Golden-angle Radial Sampling for Fast and Flexible Dynamic Volumetric MRI,” *Magnetic Resonance in Medicine*, vol. 72, no. 3, pp. 707–717, Oct 2013. [page 90]
- [349] H. Haji-Valizadeh, A. A. Rahsepar, J. D. Collins, E. Bassett, T. Isakova, T. Block, G. Adluru, E. V. R. DiBella, D. C. Lee, J. C. Carr, and D. Kim, “Validation of Highly Accelerated Real-time Cardiac Cine MRI with Radial K-space Sampling and Compressed Sensing in Patients at 1.5T and 3T,” *Magnetic Resonance in Medicine*, vol. 79, no. 5, pp. 2745–2751, Sep 2017. [page 90]
- [350] N. Vaswani and J. Zhan, “Recursive Recovery of Sparse Signal Sequences From Compressive Measurements: A Review,” *IEEE Transactions on Signal Processing*, vol. 64, no. 13, pp. 3523–3549, July 2016. [page 90]

- [351] C. A. Metzler, A. Maleki, and R. G. Baraniuk, “From Denoising to Compressed Sensing,” *IEEE Transactions on Information Theory*, vol. 62, no. 9, pp. 5117–5144, Sept 2016. [page 90]
- [352] R. Wang, Z. Yang, L. Liu, J. Deng, and F. Chen, “Decoupling Noise and Features via Weighted ℓ_1 -analysis Compressed Sensing,” *ACM Transactions on Graphics*, vol. 33, no. 2, pp. 18:1–18:12, Apr 2014. [page 90]
- [353] W. Lu and N. Vaswani, “Modified Basis Pursuit Denoising (modified-BPDN) for Noisy Compressive Sensing with Partially Known Support,” in *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, March 2010, pp. 3926–3929. [page 90]
- [354] T. Goldstein and S. Osher, “The Split Bregman Method for L1-Regularized Problems,” *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp. 323–343, Apr 2009. [page 90]
- [355] A. Beck and M. Teboulle, “A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems,” *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, March 2009. [page 90]
- [356] M. Rostami, O. Michailovich, and Z. Wang, “Image Deblurring Using Derivative Compressed Sensing for Optical Imaging Application,” *IEEE Transactions on Image Processing*, vol. 21, no. 7, pp. 3139–3149, July 2012. [page 90]
- [357] W. Dong, L. Zhang, G. Shi, and X. Wu, “Image Deblurring and Super-Resolution by Adaptive Sparse Domain Selection and Adaptive Regularization,” *IEEE Transactions on Image Processing*, vol. 20, no. 7, pp. 1838–1857, July 2011. [page 90]
- [358] J. Cai, S. Osher, and Z. Shen, “Linearized Bregman Iterations for Frame-Based Image Deblurring,” *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 226–252, March 2009. [page 90]
- [359] D. Krishnan, T. Tay, and R. Fergus, “Blind Deconvolution Using a Normalized Sparsity Measure,” in *2011 IEEE Conference on Computer Vision and Pattern Recognition*, June 2011, pp. 233–240. [page 90]
- [360] R. F. Marcia and R. M. Willett, “Compressive Coded Aperture Superresolution Image Reconstruction,” in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, March 2008, pp. 833–836. [page 90]
- [361] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang, “Coupled Dictionary Training for Image Super-Resolution,” *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3467–3478, Aug 2012. [page 90]

- [362] Y. Rivenson, A. Stern, and B. Javidi, “Single Exposure Super-resolution Compressive Imaging by Double Phase Encoding,” *Optics Express*, vol. 18, no. 14, pp. 15 094–15 103, Jul 2010. [page 90]
- [363] X. Gao, K. Zhang, D. Tao, and X. Li, “Image Super-Resolution With Sparse Neighbor Embedding,” *IEEE Transactions on Image Processing*, vol. 21, no. 7, pp. 3194–3205, July 2012. [page 90]
- [364] M. Hirsch, G. Wetzstein, and R. Raskar, “A Compressive Light Field Projection System,” *ACM Transactions on Graphics*, vol. 33, no. 4, pp. 58:1–58:12, Jul 2014. [page 90]
- [365] A. Maimone, G. Wetzstein, M. Hirsch, D. Lanman, R. Raskar, and H. Fuchs, “Focus 3D: Compressive Accommodation Display,” *ACM Transactions on Graphics*, vol. 32, no. 5, pp. 153:1–153:13, Oct 2013. [page 90]
- [366] Y. Huo, R. Wang, S. Jin, X. Liu, and H. Bao, “A Matrix Sampling-and-recovery Approach for Many-lights Rendering,” *ACM Transactions on Graphics*, vol. 34, no. 6, pp. 210:1–210:12, Oct 2015. [page 90]
- [367] 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,” *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 765–776, Jul 2005. [pages 91 and 92]
- [368] 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 92]
- [369] M. Hirsch, S. Sivaramakrishnan, S. Jayasuriya, A. Wang, A. Molnar, R. Raskar, and G. Wetzstein, “A switchable Light Field Camera Architecture with Angle Sensitive Pixels and Dictionary-based Sparse Coding,” in *2014 IEEE International Conference on Computational Photography (ICCP)*, May 2014, pp. 1–10. [pages 92 and 93]
- [370] M. Levoy and P. Hanrahan, “Light Field Rendering,” in *Proceedings of SIGGRAPH 1996*. ACM, 1996, pp. 31–42. [page 92]
- [371] J. Unger, S. Gustavson, P. Larsson, and A. Ynnerman, “Free Form Incident Light Fields,” in *Proceedings of the Nineteenth Eurographics Conference on Rendering*, ser. EGSR ’08. Eurographics Association, 2008, pp. 1293–1301. [page 92]

- [372] R. Ng, M. Levoy, M. Bredif, G. Duval, M. Horowitz, and P. Hanrahan, “Light Field Photography with a Handheld Plenoptic Camera,” *Technical Report CTSR 2005-02, Stanford University*, 2005. [page 92]
- [373] A. Lumsdaine and T. Georgiev, “The focused plenoptic camera,” in *2009 IEEE International Conference on Computational Photography (ICCP)*, April 2009, pp. 1–8. [page 92]
- [374] T.-J. Li, S. Li, Y. Yuan, Y.-D. Liu, C.-L. Xu, Y. Shuai, and H.-P. Tan, “Multi-focused Microlens Array Optimization and Light Field Imaging Study Based on Monte Carlo Method,” *Optics Express*, vol. 25, no. 7, pp. 8274–8287, Apr 2017. [page 92]
- [375] A. Wang, P. Gill, and A. Molnar, “Light Field Image Sensors Based on the Talbot Effect,” *Applied optics*, vol. 48, no. 31, pp. 5897–5905, Nov 2009. [page 93]
- [376] A. Wang, P. R. Gill, and A. Molnar, “An Angle-sensitive CMOS Imager for Single-sensor 3D Photography,” in *2011 IEEE International Solid-State Circuits Conference*, Feb 2011, pp. 412–414. [page 93]
- [377] 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. [pages 93 and 95]
- [378] C.-K. Liang, T.-H. Lin, B.-Y. Wong, C. Liu, and H. H. Chen, “Programmable Aperture Photography: Multiplexed Light Field Acquisition,” *ACM Transactions on Graphics*, vol. 27, no. 3, pp. 55:1–55:10, Aug 2008. [pages 93 and 95]
- [379] Z. Xu and E. Y. Lam, “A High-resolution Lightfield Camera with Dual-mask Design,” pp. 1–11, 2012. [page 93]
- [380] 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 96, 97, and 98]
- [381] T. Weyrich, J. Lawrence, H. P. A. Lensch, S. Rusinkiewicz, and T. Zickler, “Principles of Appearance Acquisition and Representation,” *Foundations and Trends in Computer Graphics and Vision*, vol. 4, no. 2, pp. 75–191, Aug 2009. [page 98]
- [382] M. Kurt and D. Edwards, “A Survey of BRDF Models for Computer Graphics,” *SIGGRAPH Comput. Graph.*, vol. 43, no. 2, pp. 4:1–4:7, May 2009. [page 98]

- [383] E. Veach and L. Guibas, “Bidirectional Estimators for Light Transport,” in *Photorealistic Rendering Techniques*. Springer Berlin Heidelberg, 1995, pp. 145–167. [page 98]
- [384] S. Popov, R. Ramamoorthi, F. Durand, and G. Drettakis, “Probabilistic Connections for Bidirectional Path Tracing,” in *Proceedings of the 26th Eurographics Symposium on Rendering*, ser. EGSR ’15. Eurographics Association, 2015, pp. 75–86. [page 98]
- [385] H. W. Jensen, “Global Illumination Using Photon Maps,” in *Proceedings of the Eurographics Workshop on Rendering Techniques ’96*. Springer-Verlag, 1996, pp. 21–30. [page 98]
- [386] A. S. Kaplanyan and C. Dachsbacher, “Adaptive Progressive Photon Mapping,” *ACM Transactions on Graphics*, vol. 32, no. 2, pp. 16:1–16:13, Apr 2013. [page 98]
- [387] H. Qin, X. Sun, Q. Hou, B. Guo, and K. Zhou, “Unbiased Photon Gathering for Light Transport Simulation,” *ACM Transactions on Graphics*, vol. 34, no. 6, pp. 208:1–208:14, Oct 2015. [page 98]
- [388] E. Veach and L. J. Guibas, “Metropolis Light Transport,” in *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH ’97)*. ACM Press/Addison-Wesley Publishing Co., 1997. [page 98]
- [389] T.-M. Li, J. Lehtinen, R. Ramamoorthi, W. Jakob, and F. Durand, “Anisotropic Gaussian Mutations for Metropolis Light Transport Through Hessian-Hamiltonian Dynamics,” *ACM Transactions on Graphics*, vol. 34, no. 6, pp. 209:1–209:13, Oct 2015. [page 98]
- [390] M. Manzi, F. Rousselle, M. Kettunen, J. Lehtinen, and M. Zwicker, “Improved Sampling for Gradient-domain Metropolis Light Transport,” *ACM Transactions on Graphics*, vol. 33, no. 6, pp. 178:1–178:12, Nov 2014. [page 98]
- [391] J. Pantaleoni, “Charted Metropolis Light Transport,” *ACM Transactions on Graphics*, vol. 36, no. 4, pp. 75:1–75:14, Jul 2017. [page 98]
- [392] M. Zwicker, W. Jarosz, J. Lehtinen, B. Moon, R. Ramamoorthi, F. Rousselle, P. Sen, C. Soler, and S.-E. Yoon, “Recent Advances in Adaptive Sampling and Reconstruction for Monte Carlo Rendering,” *Computer Graphics Forum*, vol. 34, no. 2, pp. 667–681, June 2015. [page 98]
- [393] P. Bauszat, M. Eisemann, and M. Magnor, “Guided Image Filtering for Interactive High-quality Global Illumination,” *Computer Graphics Forum*, vol. 30, no. 4, pp. 1361–1368, July 2011. [page 98]

- [394] B. Moon, J. Y. Jun, J. Lee, K. Kim, T. Hachisuka, and S.-E. Yoon, “Robust Image Denoising Using a Virtual Flash Image for Monte Carlo Ray Tracing,” *Computer Graphics Forum*, vol. 32, no. 1, pp. 139–151, July 2013. [page 98]
- [395] F. Rousselle, C. Knaus, and M. Zwicker, “Adaptive Rendering with Non-local Means Filtering,” *ACM Transactions on Graphics*, vol. 31, no. 6, pp. 195:1–195:11, Nov 2012. [page 98]
- [396] M. Delbracio, P. Musé, A. Buades, J. Chauvier, N. Phelps, and J.-M. Morel, “Boosting Monte Carlo Rendering by Ray Histogram Fusion,” *ACM Transactions on Graphics*, vol. 33, no. 1, pp. 8:1–8:15, Feb 2014. [page 98]
- [397] S. Bako, T. Vogels, B. Mcwilliams, M. Meyer, J. Novák, A. Harvill, P. Sen, T. Deroose, and F. Rousselle, “Kernel-predicting Convolutional Networks for Denoising Monte Carlo Renderings,” *ACM Transactions on Graphics*, vol. 36, no. 4, pp. 97:1–97:14, Jul 2017. [page 98]
- [398] P. Sen and S. Darabi, “Compressive Rendering: A Rendering Application of Compressed Sensing,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 4, pp. 487–499, Apr 2011. [pages 99, 101, and 102]
- [399] B. Choudhury, R. Swanson, F. Heide, G. Wetzstein, and W. Heidrich, “Consensus Convolutional Sparse Coding,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017, pp. 4290–4298. [page 102]
- [400] J. Liu, P. Musialski, P. Wonka, and J. Ye, “Tensor Completion for Estimating Missing Values in Visual Data,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 208–220, Jan 2013. [page 102]
- [401] 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. [page 102]
- [402] G. Yu, G. Sapiro, and S. Mallat, “Solving Inverse Problems With Piecewise Linear Estimators: From Gaussian Mixture Models to Structured Sparsity,” *IEEE Transactions on Image Processing*, vol. 21, no. 5, pp. 2481–2499, May 2012. [page 102]
- [403] M. Zhou, H. Chen, J. Paisley, L. Ren, L. Li, Z. Xing, D. Dunson, G. Sapiro, and L. Carin, “Nonparametric Bayesian Dictionary Learning for Analysis of Noisy and Incomplete Images,” *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 130–144, Jan 2012. [page 102]
- [404] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, “Deconvolutional Networks,” in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2010, pp. 2528–2535. [page 102]

- [405] H. Bristow, A. Eriksson, and S. Lucey, “Fast Convolutional Sparse Coding,” in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, June 2013, pp. 391–398. [page 102]
- [406] B. Wohlberg, “Efficient Algorithms for Convolutional Sparse Representations,” *IEEE Transactions on Image Processing*, vol. 25, no. 1, pp. 301–315, Jan 2016. [page 102]
- [407] F. Heide, W. Heidrich, and G. Wetzstein, “Fast and Flexible Convolutional Sparse Coding,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 5135–5143. [page 102]
- [408] J. Sulam, V. Pappas, Y. Romano, and M. Elad, “Multilayer Convolutional Sparse Modeling: Pursuit and Dictionary Learning,” *IEEE Transactions on Signal Processing*, vol. 66, no. 15, pp. 4090–4104, Aug 2018. [page 102]
- [409] G. Eilertsen, P. Larsson, and J. Unger, “A Versatile Material Reflectance Measurement System for Use in Production,” in *SIGRAD 2011*. Linköping University Electronic Press, 2011. [page 105]
- [410] F. Nicodemus, J. C. Richmond, J. J. Hsia, I. W. Ginsberg, and T. L. Limperis, “Geometrical Considerations and Nomenclature for Reflection,” vol. 160, 10 1977. [page 104]
- [411] R. L. Cook and K. E. Torrance, “A Reflectance Model for Computer Graphics,” *ACM Transactions on Graphics*, vol. 1, no. 1, pp. 7–24, Jan 1982. [page 105]
- [412] X. D. He, K. E. Torrance, F. X. Sillion, and D. P. Greenberg, “A Comprehensive Physical Model for Light Reflection,” *SIGGRAPH Computer Graphics*, vol. 25, no. 4, pp. 175–186, Jul 1991. [page 105]
- [413] J. Lawrence, S. Rusinkiewicz, and R. Ramamoorthi, “Efficient BRDF Importance Sampling Using a Factored Representation,” *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 496–505, Aug 2004. [page 105]
- [414] A. Bilgili, A. Öztürk, and M. Kurt, “A General BRDF Representation Based on Tensor Decomposition,” *Computer Graphics Forum*, vol. 30, no. 8, 2011. [page 105]
- [415] G. J. Ward, “Measuring and Modeling Anisotropic Reflection,” *SIGGRAPH Computer Graphics*, vol. 26, no. 2, pp. 265–272, Jul 1992. [page 105]
- [416] M. Ashikhmin and P. Shirley, “An Anisotropic Phong BRDF Model,” *Journal of Graphics Tools*, vol. 5, no. 2, pp. 25–32, Feb 2000. [page 105]

- [417] M. M. Stark, J. Arvo, and B. Smits, “Barycentric Parameterizations for Isotropic BRDFs,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 11, no. 2, pp. 126–138, March 2005. [page 107]
- [418] S. Hajisharif, J. Kronander, and J. Unger, “HDR Reconstruction for Alternating Gain (ISO) Sensor Readout,” in *Eurographics 2014 Short Papers*, May 2014. [page 108]
- [419] S. Hajisharif, J. Kronander, and J. Unger, “Adaptive DualISO HDR Reconstruction,” *EURASIP Journal on Image and Video Processing*, vol. 2015, no. 41, Dec 2015. [page 108]
- [420] 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 108]
- [421] M. Elad, “Sparse and redundant representation modeling—what next?” *IEEE Signal Processing Letters*, vol. 19, no. 12, pp. 922–928, Dec 2012. [page 111]
- [422] J. B. Nielsen, H. W. Jensen, and R. Ramamoorthi, “On Optimal, Minimal BRDF Sampling for Reflectance Acquisition,” *ACM Transactions on Graphics*, vol. 34, no. 6, pp. 186:1–186:11, Oct 2015. [page 116]

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-152863>