Spectrally-Programmable Cameras for Imaging and Inference

Submitted in partial fulfillment of the requirements for

the degree of

Doctor of Philosophy

in

Department of Electrical and Computer Engineering

Vishwanath Saragadam

B.Tech. (Hons.), Electrical Engineering, Indian Institute of Technology Madras

M.S., Electrical and Computer Engineering, Carnegie Mellon University

Carnegie Mellon University

Pittsburgh, PA

May 2020

Acknowledgements

I consider myself extremely lucky to have Profs. Aswin Sankaranarayanan and Xin Li as my advisers, who will be mentors for life. From the very beginning of this journey, both have given me ample freedom to do my own research and to think grand. I have benefited from Aswin's push towards deep insights, and Xin's strong emphasis on practical solutions. Most importantly, both my mentors went beyond graduate studies to advise me in my professional career. It is hence no surprise that the first set of people to take help about a tough decision are Aswin and Xin, be it internships, or a career in academia.

I cannot thank Aswin enough for all his guidance he has provided, but I will make an effort. I can captured Aswin's attitude to research in one simple sentence, "Nail it". He always motivated me to think hard about problems, and get a deep understanding of even the simplest of observations in the lab. Aswin painstakingly helped me debug software and hardware problems, and everytime he did, I learned at least one thing new. Aswin is also one of the best teachers I have met in my life. To this end, I often sat in his classes just to learn about effective ways of teaching. He is extremely patient in his lectures and always encourages students to ask questions, traits that I have tried to emulate in my own recitation sessions. This carries over in his research guidance as well. Aswin taught me how to write good papers, and deliver a well formulated presentation – both which will hold me in good stead in my future endeavors. Perhaps the best lesson he taught me was, "all you need is just one dB of improvement a day. In a month, you will be 30dB better", stating the importance of consistency and perseverance.

I am thankful to Jose Moura, Srinivasa Narasimhan, Yuejie Chi, and Richard Baraniuk for actively shaping my thesis into an impactful contribution. They ensured that every idea I listed as part of the thesis came together into a coherent story. I am ever grateful to them for their inputs during proposal as well as defense presentations.

If it is true that we are averages of our peers and colleagues, I must be very lucky, considering how each of my labmates have exemplary qualities. Jian taught me whatever I know about optics, and continues to be a collaborator. Harry taught me how to use chopsticks, and set a high bar for thoroughness of results. Chia-Yin showed me that life is a journey to enjoy, and has always been a bubbly presence. Yi Hua constantly reminded me that my notations are not clear, and contributed to many of the schematics in my papers. Anqi patiently helped me measure spectral profiles in biting cold, including driver her car close to the curb so that I could reliably make measurements. Rick has taught me that the best ideas are the simplest, and has been the funkiest late-night lab partner. Byeongjoo, Wei-Yu, Kuldeep, Bhargav, Tejas, all made my PhD life more exciting, with our never ending conversations about various topics (to all out there, how's life!)

My graduate life was made possible with the strong support of my friend circle. Dhivya Eswaran, who works on graph mining, taught me that debugging hardware should be the same as debugging software, and that the depth of curiosity is as important as breadth. Nothing has helped me more in my lab experiments than this insight. Praveen Venkatesh is a wonderful house mate, and it was a joy spending long hours discussing reinterpretations of the sampling theorem, the elements of good teaching, or the importance of impactful research. He is another amazing teacher, who I always go to for clarification on how to present a topic. Ranjani Srinivasan taught me the importance of understanding other's perspective, and the joy of long philosophical discussions. It is no understatement when I say that my PhD would not have been possible without them.

This thesis was practically born out of reading groups. I learned about Krylov subspace techniques in a fall reading group where we were discussed about optical light transport operators. I learned about diffraction due to lenses and apertures in a summer reading group on Fourier optics that I co-organized with my lab mates. To this end, I am indebted to everyone who I had several fruitful discussions, including Satwik Kottur, Alankar Kotwal, Adithya Pediredla, Chao Liu, Arun Nair (who patiently read my derivations for the space-spectrum uncertainty product) and many others.

This dissertation was supported by the CIT Dean's tuition fellowship, the Prabhu and Poonam Goel graduate fellowship, NSF CAREER grant CCF-1652569, National Geospatial Agency's Academic Research Program (Award no. HM0476-17-1-2000), Intel ISRA on compressive sensing, and NSF Expeditions grant 1730147. I gratefully acknowledge their support.

I want to take this opportunity to thank the unsung heroes in my graduate life, Carolyn Patterson, Claire Bauerle, and Allison Ervin, who helped me all the logistics and administrative tasks. Allison Ervin made my early mornings more joyful, when I would barge into her office and we would discuss about life at CMU, visiting other places, or our common interest, celery.

Last, but not the least, my mother, father and brother have given me enormous support. I consider myself lucky to have a mother who instills curiosity, and a father who instills hard work. But most importantly, both of them encouraged me to pursue my interests without any reservations. In a sense, whatever I am today is largely because of their love and support.

Abstract

Understanding light and its interactions with materials in a scene forms the bedrock of modern computer vision. In this context, the spectral properties of light play a very important role, especially when we seek to study the material composition of a scene. As a consequence, spectral measurements find applications across a wide range of scientific fields such as medical diagnostics, microscopy, geospace intelligence, remote sensing, and computer vision.

Several vision tasks benefit immensely from capturing spectra at all spatial locations in a scene. This requires an optical system called hyperspectral camera, which captures images across finely spaced wavelengths. Despite its wide applicability, measuring a high resolution hyperspectral image is inherently a hard task. Sampling a scene over million of spatial locations, and across hundreds of spectral bands results in diminishing photon count at each spatio-spectral voxel, leading to extremely low signal to noise ratios (SNR). This is often compensated with long exposure times, which precludes imaging of dynamic scenes. Further, the giga-pixels of data associated with each scan places immense burden on capture and processing hardware.

The work in this thesis seeks to simplify the process of capturing spectral information of a scene with design of novel imaging systems. This thesis relies on two key observations. First, despite the high dimensional nature of hyperspectral images, the number of distinct materials in any given scene is very small; this leads to a concise low-dimensional representation of the hyperspectral image. Second, owing to this low diversity, capturing a small set of spectrally-filtered images of the scene suffices for most sensing and inference tasks. Exploiting these two observations, this thesis builds novel and efficient optical systems for imaging and inference.

Central to the contributions of this thesis is an optical system that can provide programmable spectral filtering, by attenuating intensity of light at each wavelength arbitrarily and capturing the resultant image. The first contribution of this thesis shows that capturing sharp images with arbitrarily high resolution spectral filtering is not possible – a property that arises due to the shape of the pupil function of the camera. This fundamental limit is provided in the form of the space-spectrum uncertainty principle, which sets a lower bound on product of spectral and spatial spreads. We then show that the resolutions can be enhanced computationally, if the pupil function is carefully engineered to introduce invertible spatial and spectral blurs.

Armed with the insights of a spectrally-programmable setup, we show that such cameras can be used to efficiently sense hyperspectral images. Since the true complexity of sensing hyperspectral images lies not in high resolution space or spectrum, but only the diversity of materials, the hyperspectral image can

be represented using a low-rank matrix model. This thesis provides a novel *adaptive sensing* strategy to *optically compute* this low-rank model. We note that the dominant spatial and spectral singular vectors can be sensed by building two optical operators, namely a spatially-coded spectrometer, and a spectrally-programmable camera. By alternating between the two operators, and using output of one operator as input to the second, we can measure a low-rank approximation with as few as ten measurements – contrasted with several hundreds of measurements for fully scanning the hyperspectral image.

Finally, the thesis builds on spectral-programmability and optical computing to enable per-pixel material classification. This is achieved by capturing images of the scene with *learned, discriminative spectral filters* and then using the images to classify materials. This enables a per-pixel classification strategy with a small set of high SNR measurements – thereby leading to real-time vision capabilities. At its culmination, this thesis lays groundwork for making hyperspectral cameras more practical by introducing computing into the sensing pipeline, and moving most of computational burden into the optical domain. This successfully decouples the number of measurements and SNR, thereby allowing future optical systems to achieve very high resolution along spatial and spectral axes.

For Dad, and Mom

Contents

List of Figures xiii

List of Tables xvi

List of Tables xvi

1 Introduction 1

1.1 Conventional Hyperspectral Camera Architecture . . . 2

1.2 Compressive Sensing . . . 4

1.3 Low Rank Model for Hyperspectral Images . . . 5

1.3.1 Optical Computing of HSIs . . . 6

1.3.2 Spectrally-programmable cameras . . . 7

1.3.3 Challenges with spectrally-programmable cameras . . . 7

1.4 Thesis Contributions . . . 8

1.5 Thesis Roadmap . . . 10

1.6 Impact of the proposed research . . . 11

2 Space-spectrum Uncertainty Analysis 13

2.1 Introduction . . . 13

2.1.1 Problem setting . . . 13

2.1.2 Main result . . . 14

2.1.3 Implications . . . 15

2.2 Prior Work . . . 16

2.3 Fundamental Limits of Spatial/Spectral Resolution . . . 18

2.3.1 The space-spectrum uncertainty principle . . . 21

2.3.2 Verification using simulations . . . 23

2.4 Experiments . . . 23
2.4.1 Spectral programming . . . 25
2.5 Conclusion . . . 29

3 KRISM **31**
3.1 Introduction . . . 31
3.2 Prior work . . . 33
3.3 Optical Krylov Subspaces for Hyperspectral Imaging . . . 37
3.4 Coded apertures for simultaneous sensing of space and spectrum . . . 39
3.4.1 Optical setup . . . 39
3.4.2 Failure of slits and open apertures . . . 42
3.4.3 Design of aperture codes . . . 43
3.4.4 Other code designs . . . 45
3.4.5 Performance comparison . . . 48
3.4.6 Spectral deconvolution . . . 48
3.4.7 Spatial deconvolution . . . 49
3.5 Synthetic experiments . . . 49
3.6 The KRISM Optical setup . . . 53
3.6.1 Choice of code size . . . 56
3.6.2 Handling positive/negative data . . . 57
3.6.3 Calibrating the optical setup . . . 57
3.7 Real Experiments . . . 62
3.8 Discussion and conclusion . . . 66

4 Programmable Spectrometry **69**
4.1 Introduction . . . 69
4.2 Prior Work . . . 71
4.2.1 Spectral Classification . . . 72
4.2.2 Hyperspectral classification . . . 72
4.2.3 Need for high spectral resolution . . . 73
4.2.4 Sensing for classification . . . 74
4.2.5 Optical computing . . . 74
4.2.6 Dynamic spectral filters . . . 74

4.3 Programmable Spectral Filter . . . 75
4.3.1 4f system for spectral programming . . . 76
4.3.2 Optical setup . . . 76
4.3.3 Effect of coded aperture . . . 76
4.4 Learning Discriminant Filters . . . 77
4.4.1 Support Vector Machine . . . 78
4.4.2 Deep Neural Networks . . . 78
4.4.3 Simulations . . . 79
4.5 Experiments . . . 79
4.5.1 Learning details . . . 80
4.5.2 Optical setup . . . 81
4.5.3 Handling illumination . . . 85
4.5.4 Dataset . . . 85
4.5.5 Training classifiers . . . 85
4.5.6 Handling scale of features . . . 86
4.5.7 Accuracy vs. number of filters . . . 87
4.5.8 Accuracy vs. smoothness . . . 87
4.5.9 Binary classification . . . 87
4.5.10 Remote photoplethismography . . . 89
4.5.11 Multi-class classification with DNNs . . . 89
4.5.12 Discussion . . . 92
4.5.13 Limitations . . . 93
4.6 Conclusion . . . 94

5 Conclusions **95**

A Supplementary for KRISM **97**

A.1 Real experiments . . . 97
A.1.1 Visualizing spatial images . . . 97
A.1.2 Comparison of singular values and singular vectors . . . 98
A.2 Synthetic experiments . . . 98
A.2.1 Performance with high spectral resolution . . . 99
A.2.2 Performance with low spectral resolution . . . 100

B Supplementary for Programmable Spectrometry **109**

B.1 Simulations 109

Bibliography **113**

List of Figures

1 Introduction **1**

1.1 Hyperspectral imaging applications . . . 2
1.2 Hyperspectral camera architectures . . . 3
1.3 Optical setup for capturing images with spectral modulation . . . 8
1.4 Thesis layout . . . 9

2 Space-spectrum Uncertainty Analysis of Coded Aperture Systems **13**

2.1 Optical setup for capturing images with spectral programming . . . 14
2.2 Simulated $x - \lambda$ blur for Gaussian window . . . 22
2.3 Simulations on common aperture shapes . . . 23
2.4 Schematic and image of our lab prototype . . . 24
2.5 Visualization of spectral and spatial resolutions . . . 25
2.6 Quantitative measurement of resolutions . . . 25
2.7 Schematic and image of prototype for spectral programming . . . 26
2.8 Spectral programming with narrowband sources . . . 27
2.9 Effect of narrowband spectral programming on spatial resolutio . . . 28

3 KRISM – Krylov Subspace-based Optical Computing of Hyperspectral Images **31**

3.1 Optical computing for high resolution hyperspectral images. . . . 32
3.2 Concise representation of HSIs with a low-rank model. . . . 33
3.3 Schematic for simultnenous spatio-spectral measurements with a coded aperture. . . . 38
3.4 Effect of pupil code on spatial and spectral resolutions. . . . 41

3.5 Visualization of optimally engineered pupil codes. . . . 42
3.6 Frequency response of spatial and spectral blur for various pupil codes. . . . 45
3.7 Comparison of performance of various codes. . . . 46
3.8 Calibration results with narrowband spectral filters. . . . 47
3.9 Spectral profiles of common illuminants measured with lab prototype. . . . 47
3.10 Deconvolution results for a Siemen star target. . . . 48
3.11 Modulation Transfer Function of the lab prototype. . . . 50
3.12 Evaluation against snapshot techniques. . . . 51
3.13 Evaluation against multi-frame techniques. . . . 52
3.14 Schematic of KRISM optical setup. . . . 53
3.15 Photograph of KRISM lab prototype. . . . 53
3.16 List of components for the lab prototype. . . . 55
3.17 Implementation of spatial/spectral codes with positive and negative components. . . . 57
3.18 Camera-SLM calibration for lab prototype. . . . 59
3.19 Calibration of code and wavelengths. . . . 60
3.20 Calibration of spectral response. . . . 60
3.21 Visulaization of Krylov iterations. . . . 62
3.22 Comparison of captured singular vectors. . . . 63
3.23 Real data captured with our optical setup. . . . 65
3.24 Verification with Macbeth color chart. . . . 66
3.25 Failure for higher singular vectors. . . . 67

4 Programmable Spectrometry – Per-pixel Material Classification using Learned Spectral Filters 69

4.1 Spectrum-based material classifier. . . . 70
4.2 Collection of spectral profiles of some everyday materials. . . . 70
4.3 Effect of resolution on spectral classification. . . . 73
4.4 Setup schematic for implementing programmable spectral filters. . . . 75
4.5 Implementing spectral filters. . . . 77
4.6 Neural network architecture for spectral classification. . . . 78
4.7 Comparisons of various classification architectures. . . . 79
4.8 Image of lab prototype marked with components. . . . 82

4.9 Example HSI scanned with lab prototype. . . . 82
4.10 Wavelength calibration. . . . 83
4.11 Coding strategy for implementing spectral filters. . . . 83
4.12 Spatial deconvolution results. . . . 84
4.13 Learned neural network spectral filters. . . . 86
4.14 Accuracy as a function of number of spectral filters. . . . 87
4.15 Accuracy vs. smoothness of filters. . . . 88
4.16 Spatial resolution of binary classifier. . . . 88
4.17 Results with matched filters. . . . 89
4.18 Results with SVM classifier. . . . 90
4.19 SNR advantage of optical computing. . . . 90
4.20 Video rate binary classifier example. . . . 91
4.21 Results with remote pulse oximetry. . . . 91
4.22 Classification of outdoor materials. . . . 92
4.23 Comparison of optical and full-scan spectral filters. . . . 92
4.24 Multiclass classification with neural networks. . . . 93

A Supplementary for KRISM **97**

A.1 Comparison of spatial images against cellphone images. . . . 97
A.2 Comparison of spatial and spectral singular vectors measured by lab prototype. . . . 101
A.3 Images across different wavelengths . . . 102
A.4 Simulations on high resolution datasets. . . . 103
A.5 Accuracy vs. compression ratio . . . 104
A.6 KRISM vs Row/Col CS . . . 104
A.7 Mult-frame CS at high compression. . . . 105
A.8 Simulations on low resolution datasets. . . . 107
A.9 Evaluation against dicationary-based reconstruction. . . . 107
A.10 Accuracy vs. compression for low-resolution datasets . . . 108

B Supplementary for Programmable Spectrometry **109**

B.1 Simulations on existing HSI classification datasets. . . . 111

List of Tables

1.1 Comparison of hyperspectral sensing strategies 5
1.2 Low rank decomposition of Hyperspectral images 6

3.1 Comparison of HSI sensing strategies. 34
3.2 Datasets used for simulations. 50

4.1 List of materials in the spectrometry dataset. 81
4.2 Components of neural network classifier. 81

1 Introduction

We perceive the world around us by observing its interaction with light – and a fundamental and important part of this interaction is how the intensity of light varies with different wavelengths of light, called spectrum. The primary way humans observe spectrum is via colors – a three dimensional slice of the spectrum. However, going beyond colors and capturing the complete spectrum with a device called spectrometer offers unique insights about the composition of various objects in the scene and hence is critical to scientific discovery. Spectroscopy as a scientific tool was first identified by Joseph Von Fraunhoffer in 1814, when he identified the dark bands in the Sun's spectrum, which lead to the discovery of its material composition. Since then, the applications of spectral measurements has pervaded several fields of scientific discovery. It is used for accurately tracking microscopic cells and molecules by relying on fluorescence, where objects absorb light at one wavelength and emit at a different one. Raman spectroscopy seeks to estimate molecular composition by shining light on the specimen and studying the spectrum of scattered light. In medicinal sciences, it finds use in pulse oximetry, where lights of two different wavelengths are illuminated to estimate composition of oxygenated and deoxygenated blood. At macro scales, it is used for identifying and locating minerals and vegetation cover over large swathes of land efficiently, and the presence of oil slicks in oceans.

When we are interested in sampling spatial and spectral profiles, we consider the hyperspectral images (HSI). Hyperspectral cameras measure spectrum at each spatial pixel in a scene resulting in a stack of images sampled over narrowband wavelengths. The measured HSI is high dimensional with several millions of samples of spatial pixels across hundreds of spectral bands. Such a dense sampling enables the inference of material composition at each pixel, and is hence an indispensable tool in many scientific applications. The applications of HSIs are far-reaching and hence the research in this field is focused on building faster and more sensitive cameras that can capture signal at very fine spatial and spectral resolutions. There are several commercially available cameras that achieve some parts of this, either high spatial resolution, or spectral resolution, or low resolution along both axes but high frame

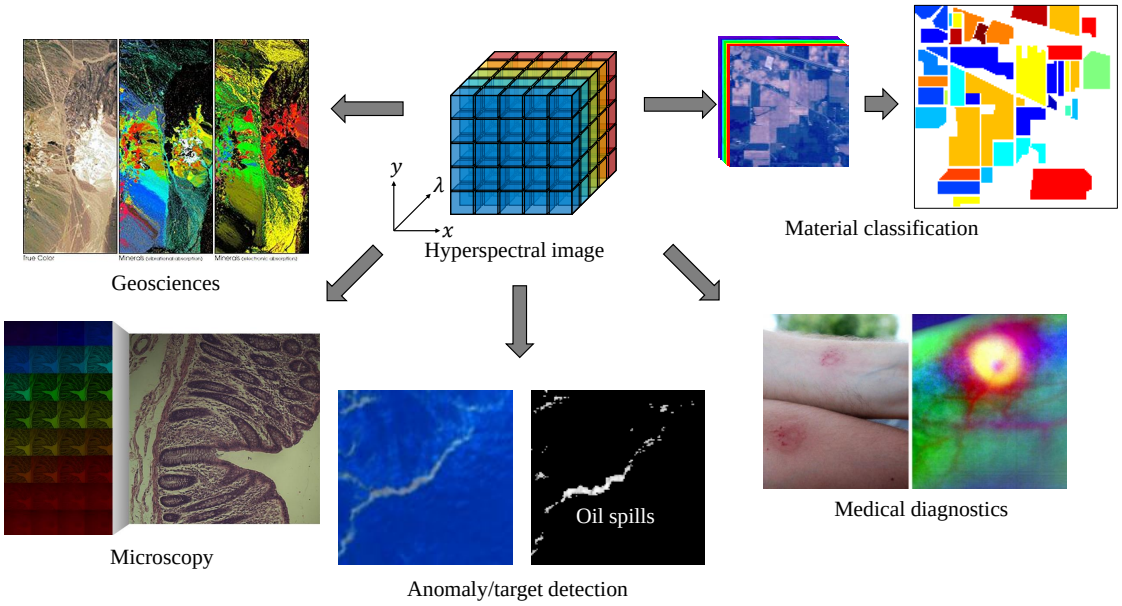


Figure 1.1: **Hyperspectral imaging applications**. Hyperspectral images find application in a wide range of fields such as geosciences[1], microscopy[2], target detection[89], medical diagnosis[3]and material classification [11].

rate. Capturing very high resolution data comes with several key challenges, which will be explained over the course of this chapter. *This thesis seeks to advance hyperspectral imaging and inference with core focus on efficient optical systems and algorithms.* To understand the need for such efficient cameras, we first need to look at the drawbacks of existing imaging systems. This will then motivate the contributions of this thesis.

1.1 Conventional Hyperspectral Camera Architecture

Commercial hyperspectral cameras come in one of the three forms (see Fig. 1.2), each with their own merits and demerits. Tunable-filter based cameras capture each narrowband image at a time, thereby scanning along the wavelength axis. Such cameras offer very high spatial resolution and are easy to build, as they require attaching the tunable filter to any existing grayscale camera. Pushbroom cameras capture spectral profiles of each row of the scene, thereby scanning along the vertical spatial axis. Push-

[1]Image credits: https://resonon.com/applications

[2]Image credits: https://thorlabs.com

[3]Image credits: https://sciencenorway.no/forskningno-hyperspectral-camera-innovation/charting-sores-and-bruises-in-multiple-colours/1411937

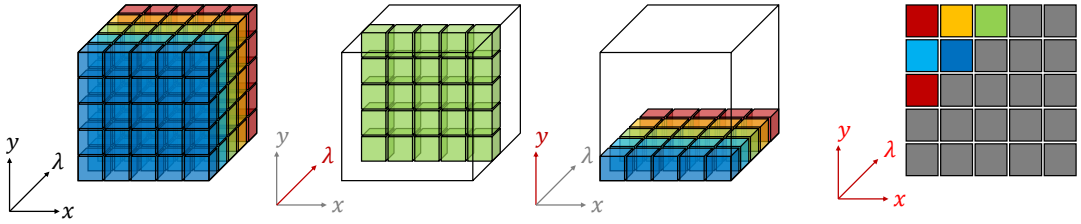


Figure 1.2: **Architecture of existing hyperspectral cameras**. Hyperspectral cameras scan image at each wavelength (tunable filter), or capture all spectral profiles of each row (pushbroom), or rely on a color field array (CFA) to capture measurements at lower resolution. In all cases, high resolution scanning requires long exposure times and fast electronic subsystems.

broom cameras come with very high spatial and spectral resolution, and are ideal for scenarios where scanning the scene involves imaging a single spatial line, such as in case of aerial imaging, or scanning objects on a conveyor belt. Both tunable-filter and pushbroom cameras are categorized as non-snapshot cameras, as they require multiple measurements of the scene to scan the complete HSI. In contrast, snapshot cameras that utilize a color field array (CFA) implement a Bayer-like pattern and require one single image. As is to be expected, such cameras trade off spatial resolution for spectral resolution and hence is not a popular choice for hyperspectral imaging.

Challenges with hyperspectral imaging. Existing hyperspectral cameras face two key challenges, and both are associated with the high dimensionality of HSIs. First, due to the fine spatio-spectral sampling, Nyquist sampling of HSIs face a severe loss in signal to noise ratio (SNR). To get an intuition, consider an HSI of 100 spectral bands, and 1000×1000 spatial pixels. Then each spatio-spectral voxel gets a billionth of the total photons entering the camera; except when imaging in extremely bright settings such as sunlight [21], such low photon count requires very long exposure times to maintain high SNR. Second, transferring such large amounts of data requires high bandwidth electronics. As an example, if each image is quantized to 8 bits, then the above mentioned camera requires 30GB/s to achieve 30 frames per second, a rate that is impractical with existing technology, and most importantly, does not scale to higher resolutions. This debilitating effect of reduced SNR and need for high data rates implies that hyperspectral cameras are not practical for dynamic scenes. Indeed, most commercially available high resolution hyperspectral cameras achieve no more than 1 frame per second[4]. However, vast number of

[4] https://www.specim.fi/afx/

scenes are inherently dynamic, such as *in vivo* imaging, life sciences imaging, and vehicular tracking. In order to tackle these challenges, we need to build systems that sense less, and are not affected by loss in SNR.

1.2 Compressive Sensing

Fortunately, HSIs are not arbitrary signals – indeed, they have concise representations – such as sparsity in wavelet domain, and low-rank structure. Compressive Sensing (CS) [9] exploits this property of visual signals to reconstruct signal that are undersampled with random linear projections. The prior information about the signal, or a signal model was used to regularize the linear inverse problem. Some common signal models for HSIs include image-wise sparsity in wavelet domain, group sparsity, and a core focus of this thesis – the low-rank approximation. This lead to a decade long research in building CS imagers for HSIs that sample compressed measurements and subsequently use signal priors to reconstruct the signal.

There are several imaging architectures for CS imaging of HSIs, and we discuss some of them here. Li *et al.* [54] achieve this via spatially-multiplexed spectral measurements, where a spatial light modulator (SLM) is used to modulate intensity of each spatial pixel, and a spectrometer is used to measure the average spectrum of the modulated scene.. Signal is then recovered with joint sparsity of all spectral bands, or by a non-negative matrix factorization method. This approach requires far fewer measurements than a full scan of the HSI, but often requires computationally complex algorithms for recovery. Spatial and spectrally-multiplexed imagers [7, 57, 58] improve upon these designs by combining both space and spectrum, which lead to better conditioning of the system. This resulted in fewer measurements, and lead to higher accuracy of results. As before, signal is recovered by exploiting sparse representations of the HSI patches in an overcomplete dictionary, which is often a time taking process.

A separate class of acquisition devices, called coded aperture snapshot spectral imagers (CASSI) [98] capture one or more [49] spectrally-smeared images. This is achieved by using a 2D mask to modulate the spatial image at all wavelengths, and then using a dispersive element such as prism or diffraction grating to spatially shift each individual spectral band.

Compressive sensing faces two main problems. First, though the sampling time is reduced, the reconstruction is often computationally complex, requiring several minutes to hours of computer time. Such systems are incapable of producing real-time results. Second, CS reconstruction faces noise-folding [5, 23], where even an oracle recovery algorithm (one that knows the exact subspace of the signal) results in a 3dB loss in reconstruction SNR for every 2× fewer measurements. This is practically an

Method	Approach	Number of measurements	Estimation accuracy under noise	Advantages	Disadvantages
Sampling	Tunable spectral filter	$N_x N_y N_\lambda$	$\sigma\sqrt{N_x N_y N_\lambda}$	Easy calibration	Low spectral resolution; high acquisition time
	Pushbroom			High spectral resolution	Optical complexity; high acquisition time
Multiplexed	Spatial multiplexing	$N_x N_y N_\lambda$	$\frac{\sigma\sqrt{N_\lambda}}{\sigma\sqrt{N_x N_y}}$	Hadamard multiplexing gain	High acquisition time
	Spectral multiplexing				
Compressive sensing	CASSI	depends on signal model		Fewer measurements	Loss in spatial/spectral resolution
	Row/column projection	$\propto k^2(N_x N_y + N_\lambda)$	[Fazel et al. 2008]		Complex optics

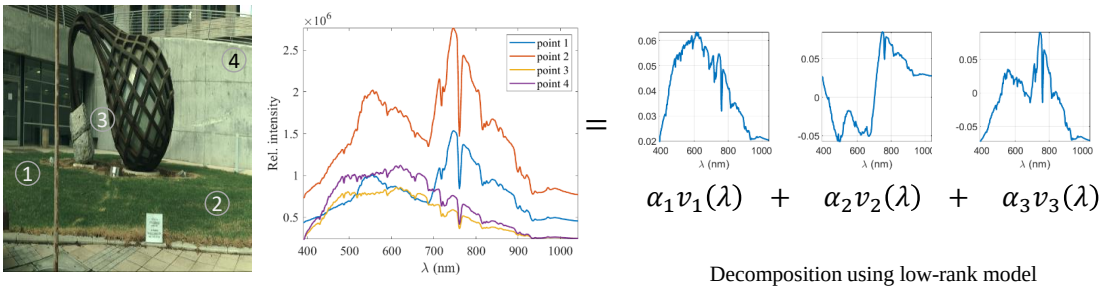
Table 1.1: **Comparison of hyperspectral sensing strategies.**

antithesis to CS – as a successful CS imager would capture data at high compression ratio – and yet this leads to *reduction* in SNR. Further, most real world signals are at best *approximated* by low-dimensional models, which adversely affects CS recovery under model mismatch. CS techniques have since been augmented with stronger signal models [10] and sampling strategies with optimized sensing matrices [79–81] which have attempted to increase SNR of reconstruction. Table 1.1 gives a broad overview of existing hyperspectral imaging technologies.

A hallmark of CS techniques is that the measurement process is non adaptive, i.e., the sensing and recovery stages are decoupled. Such non-adaptive techniques generally serve to reduce the complexity of the imaging system since there is no need for a feedback from the recovery procedure back to the camera. But, as a consequence, a significant avenue for simplifying the end-to-end system is missed. In practice, this reduces to two specific problems. One, the recovery technique is often computationally complex. As an example, recovery from CASSI measurements takes anywhere between tens of minutes with parallel processing units such as GPUs [18] to several hours [57]. Second, most priors used are domain specific and not signal specific, and hence there is a gap in the applicability of the prior to the specific instance of the signal that is being sensed. This often leads to overly smooth spectral profiles for hyperspectral imaging. We will next see how the low-dimensionality of HSIs can be exploited during the sensing phase itself.

1.3 Low Rank Model for Hyperspectral Images

Since the diversity of spectral profiles in any given scene is low, HSIs often tend to be low rank volumes, as illustrated in Fig. 1.2. If the basis spectral profiles are known, then the low-rank approximation can be



Hyperspectral image | Spectrum at selected points | Decomposition using low-rank model (small but unknown set of spectra)

Table 1.2: **Low rank decomposition of hyperspectral images**. Most HSIs can be concisely represented with a low rank model, implying that spectrum at each pixel is a combination of a small set of unknown basis spectra.

measured by capturing spectrally-filtered images of the scene, which requires far fewer measurements than either spatial or spectral resolution. Surprisingly, the same technique extends to cases when the spectral profiles are not known, and will be explained in the upcoming sections. Similarly, classification using hyper spectral images requires projection of spectrum at each pixel to a bank of spectral, thereby requiring a small number of images. Can we then design systems to directly capture these images?

1.3.1 Optical Computing of HSIs

One efficient way of sensing HSIs efficiently is to optically project its columns or rows to a low-dimensional subspace. This was explored by Park *et al.* [73], where they captured HSIs with an active illumination setup. By relying on the idea that naturally occurring materials are accurately represented using nine or fewer *known* spectral profiles [74], Park et al. showed that one can achieve video-rate results.

While Park *et al.* [73] relied on an active illumination setup, most real-world scenarios only admit passive illumination settings, where there is no control on spectral profile of the illumination. This motivates us to build a camera that allows one to modify the spectral response of the camera instead. One can utilize the low dimensional basis proposed by Parkkinen *et al.* [74] in conjunction with such a camera to potentially scan the HSI with very few measurements. The next section discusses the merits of such a programmable camera approach, and how we can generalize sensing by following an adaptive sensing approach, where the basis for spectral profiles is not fixed but estimated during sensing phase.

1.3.2 Spectrally-programmable cameras

The common technique between scanning a low rank volume and spectral classification is a spectrally-programmable camera that captures images with arbitrary spectral filters. Capturing HSIs with spectrally-programmable camera provides three important benefits. First, the number of measurements are far fewer; in case of low rank sensing, the number of measurements are commensurate to the rank of the scene's HSI, which tend to be very few [74]. Second, the spectral profiles used for either low rank sensing, or classification tend to be broadband. A direct implication of such filters is that the light levels are remarkably higher than narrowband filters used for scanning the complete HSI, leading to much higher measurement SNR. Third, the amount of post processing is minimal, as the measurements directly lead to the required data. While spectrally-programmable cameras have been proposed in the past for sensing [7, 58], they rely on random spectral projections, and then computationally expensive recovery algorithms. We will see that carefully tailoring the spectral projections in a *scene-specific manner* overcomes this computational burden.

1.3.3 Challenges with spectrally-programmable cameras

Cameras with spectral programming have been proposed in the past for various applications. Mohan *et al.* [66] proposed an "agile spectrum imager" that relied on the so-called rainbow plane to modulate spectrum of the whole scene. This was further developed by Love and Graff [63] who utilized a spatial light modulator to program the spectrum at high speeds. Such setups can be utilized for fast modulation, but are woefully inadequate when it comes to capturing or modulating at high resolutions; however, they are the *de-facto* way of modulating spectrum, and hence requires some attention. In the upcoming section, we describe the specific setup that is central to this thesis and give a broad overview of the challenge faced by spectral programming.

There are multiple ways of achieving spectral programming, but the most efficient approach involves using a diffraction grating to split light into constituent wavelengths and create the so-called rainbow plane, and then use a spatial light modulator (SLM) to achieve spectral modulation. The setup, and some brief details are provided in Fig. 1.3. The optical system consists of a series of lenses of focal length f, each subsequent pair separated by $2f$. The setup relays the image plane from plane P1 to P3 with a *pupil code* in P2. A diffraction grating placed on P3 provides a spectral dispersion of the light. The dispersed light is focused on plane P4 to form the so-called *rainbow plane*, where each point corresponds to the average intensity of light of the whole scene for a single wavelength. The image on P3 is simply relayed on to plane P5. Arbitrary spectral filtering can then be performed by placing an (SLM) on the rainbow

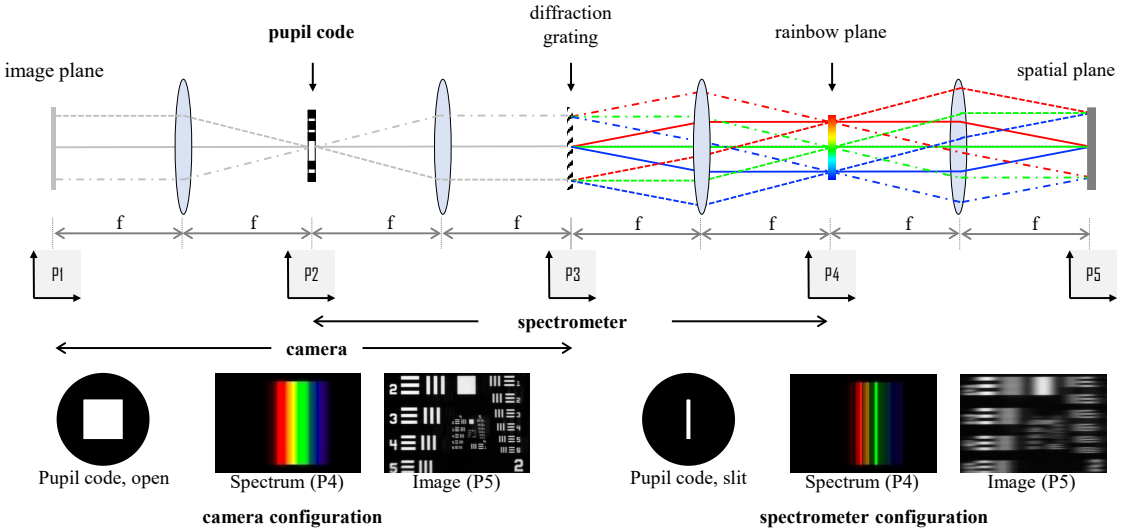


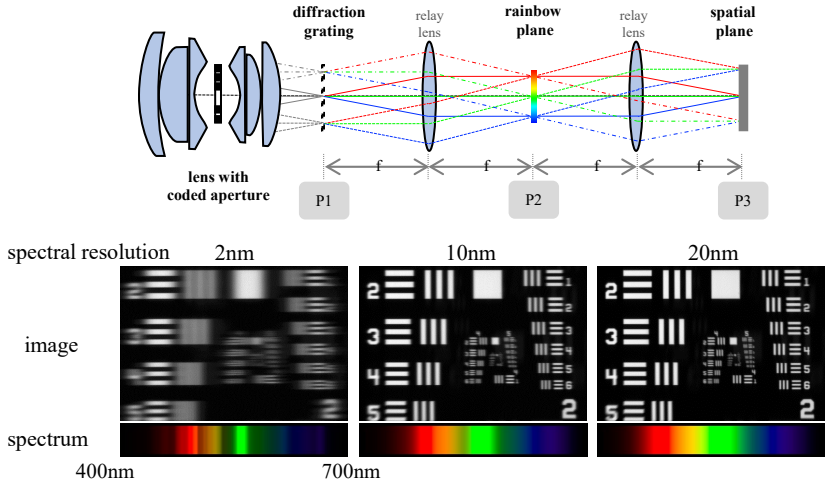
Figure 1.3: **Optical setup for capturing images with spectral modulation**. P1 is the image plane of the objective lens, P2 contains spatial frequencies of the image, where we place a pupil code, $a(x, y)$. P3 contains the image plane blurred by the aperture function. We place a diffraction grating at this plane to disperse light into different wavelengths. P4 contains the resultant spectrum and P5 is a flipped copy of P3. In a camera configuration, the pupil code consists of an open aperture and leads to sharp image but blurred spectrum. In spectrometer configuration, the pupil code is a slit and leads to sharp spectrum but blurred image. This tradeoff, methods to overcome it, and the applications is the key focus of this thesis.

plane (P4) and measuring image on plane P5.

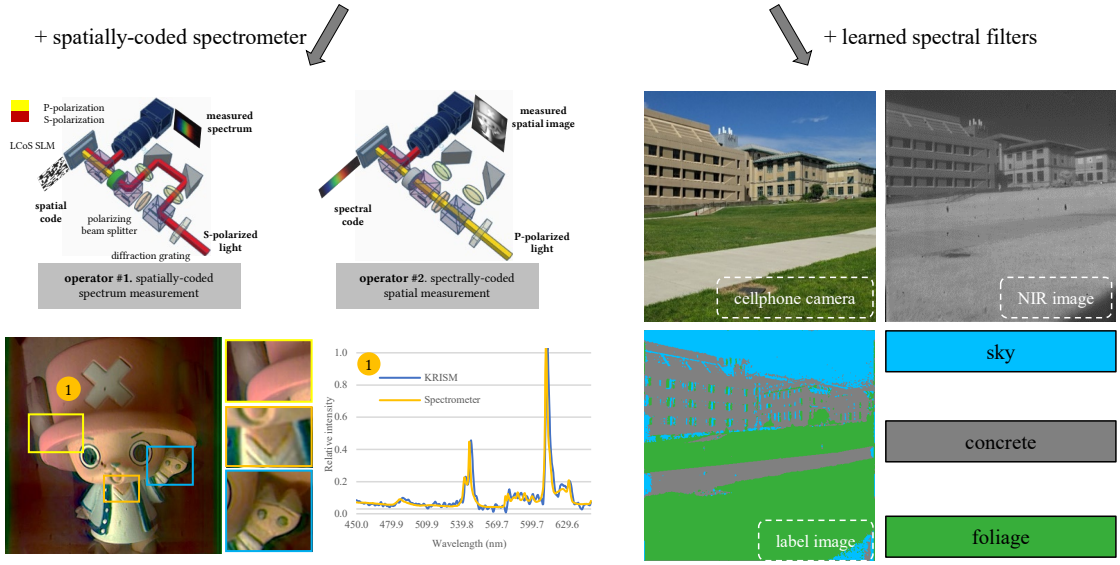
This setup will be discussed repeatedly throughout the thesis, and hence it is important to get an intuition for the role played by the pupil function right away. The optical subsystem from planes P1 to P3 is a simple camera with an aperture in its Fourier plane, and planes P2 to P4 is a spectrometer with the aperture replacing a slit. The setup provides an insight into the tradeoff between spatial and spectral resolutions. While a camera requires a large and open aperture for a compact spatial blur, this would lead to severe loss in spectral resolvability, as a spectrometer requires a narrow opening. This inherent uncertainty between spatial and spectral spread will form the background for the thesis.

1.4 Thesis Contributions

This thesis aims to solve the problem of sensing and inference of HSIs using a spectrally-programmable camera as the core technology. An overview of the thesis results is shown in Fig. 1.4. To this end, the thesis makes the following contributions:



(a) **Chapter 2:** Space-spectrum uncertainty



(b) **Chapter 3:** *Adaptive imaging* for high spatial and spectral resolution with very few measurements

(c) **Chapter 4:** *Optical computing* for per-pixel material classification

Figure 1.4: **Layout of the thesis.** We first explore (a) the space-spectrum uncertainty product in chapter 2, which states the fundamental limits of spatial and spectral resolution with a spectrally-programmable camera. Chapter 3 combines such a camera with a spatially-coded spectrometer and presents a (b) novel imaging technique for optically computing the dominant spatial and spectral singular vectors of a scene's HSI. Chapter 5 shows an important application of (c) per-pixel material classification with the programmable camera and learned spectral filters.

- *Space-spectrum uncertainty.* Spectrally-programmable cameras face a space-spectrum uncertainty, implying that it is not possible to simultaneously capture high-resolution spatial images while program-

ming the spectrum at high resolution. This phenomenon arises due to a Fourier relationship between the aperture used for obtaining spectrum and its corresponding diffraction blur in the (spatial) image. This thesis shows that the product of spatial and spectral standard deviations is lower bounded by $\frac{\lambda}{4\pi v_0}$ femto square-meters, where v_0 is the density of groves in the diffraction grating and λ is the wavelength of light. Figure 1.4 (a) show the important result of the chapter, that the images get blurred by a larger amount as spectral resolution increases.

- *Adaptive sensing of HSIs.* This thesis presents an adaptive imaging technique that optically computes a low-rank approximation of a scene's hyperspectral image, conceptualized as a matrix. Central to the proposed technique is the optical implementation of two measurement operators: a spectrally-coded imager and a spatially-coded spectrometer. By iterating between the two operators, we show that the top singular vectors of a hyperspectral image can be adaptively and optically computed with only a few iterations. This chapter also presents a pupil-code design for the spectrally-programmable camera to computationally enhance the spatial and spectral resolutions. The end product is a high resolution hyperspectral camera that requires very few measurements to capture the full HSI. A key result of this chapter is shown in Fig. 1.4 (b), where we show a captured HSI of the scene illuminated by a peaky spectrum of compact fluorescent lamp (CFL) with as few as ten measurements.

- *Optical computing for per-pixel spectral classification.* This thesis shows an important application of the spectrally-programmable camera to optically implement the spectral filtering of the scene's hyperspectral image with a set of spectral profiles to perform per-pixel material classification. This provides gains both in terms of acquisition speed — since only the relevant measurements are acquired — and in signal-to-noise ratio — since we invariably avoid narrowband filters that are light inefficient. Given training data, we use a range of classical and modern techniques including SVMs and neural networks to identify the bank of spectral profiles that facilitate material classification. Figure 1.4 (c) shows a per-pixel classification result of an outdoor scene obtained with only six spectrally filtered images. This takes hyperspectral cameras one step closer to real-time material classification which will prove to be an indispensable tool in computer vision.

1.5 Thesis Roadmap

This thesis is designed to introduce the reader to various optical systems and algorithms associated with hyperspectral cameras.

- *Chapter 2* introduces the space-spectrum uncertainty analysis for spectrally programmable cameras. The result of this chapter will be the space-spectrum bandwidth product. This chapter revisits the well known time-bandwidth product in Fourier analysis of signals and applies it to hyperspectral imaging. Along the way, the thesis will show that Gaussian-shaped pupil codes achieve theoretically minimal spectral and spatial spreads, which will also provide a clear set of guidelines for building cameras with a target spatial and/or spectral resolution.

- *Chapter 3* will focus on directly sensing a low-rank approximation of the scene's HSI with the help of a spectrally-programmable camera and a spatially-coded spectrometer. The reader will learn about Krylov iterative techniques for computing top singular vectors of a matrix, and how they can be implemented optically. This part will also provide guidelines on breaking the space-spectrum bandwidth product with computational postprocessing with engineered pupil codes for invertible spatial and spectral blurs.

- *Chapter 4* will utilize tools and ideas from chapter 2 and 3 to build a per-pixel material classification camera. There will be thorough discussion about how classification can be translated to optical domain with emphasis on choice of classifier, learning discriminating filters and a wide gamut of applications.

- *Chapter 5* will conclude the thesis with discussions about future work related to this thesis and the impact it will have on several fields in imaging and computer vision.

Each chapter is largely self-sufficient, and hence can either be read sequentially or by themselves.

1.6 Impact of the proposed research

Future technologies in imaging will be aimed at capturing very finely resolved data, across several dimensions. The thesis makes fundamental contributions to spectral programming, with emphasis on small number of high SNR, high spatial and spectral resolution measurements, with reliance on adaptive sensing, as well as optical computing. This paves way to practical and computationally inexpensive cameras for hyperspectral imaging, and will find way into more vision tasks. While the focus of this thesis was on spatial and spectral imaging, the ideas presented have far reaching impact across imaging applications, such as sampling angular variations, capturing polarization of light, and imaging at very high frame rates.

2 Space-spectrum Uncertainty Analysis of Coded Aperture Systems

Capturing spectrally-filtered images requires an optical setup that is fast, and can operate at high spatial and spectral resolutions. In this chapter, we will see that the two resolutions are tightly coupled, leading to a space-spectrum uncertainty. At the end of this chapter, we will get a good theoretical understanding of limits of spectral programming, the optimality of a Gaussian-shaped pupil, and get a brief introduction to optical systems for capturing images with arbitrary spectral filters.

2.1 Introduction

Spectrum is often a unique feature of materials and is used for identification and classification across diverse fields such as geology [19], bio-imaging [20, 56] and material identification [88, 106]. Tools such as the hyperspectral camera capture the spectrum of a scene which is subsequently used for identification and classification purposes. Capturing the full spectrum, while useful, is also wasteful especially if we are only interested in measuring similarity of the spectral profile at each pixel to a small collection of reference spectra. It is hence useful to have cameras that can *optically* perform this comparison. Such cameras, called *spectrally-programmable cameras*, have been demonstrated [63, 66] with compelling applications in computer vision. This chapter analyzes a popular design for enabling spectral programmability, and derives a fundamental relationship between its achievable spatial and spectral resolutions.

2.1.1 Problem setting

The analysis in this chapter is for the optical setup shown in Fig. 2.1, commonly used in prior art for spectral programming [58, 63, 87, 88]. The optical system consists of a series of lenses of focal length f, each subsequent pair separated by $2f$. The setup relays the image plane from plane P1 to P3 with a *pupil code* in P2. A diffraction grating placed on P3 provides a spectral dispersion of the light. The dispersed light is focused on plane P4 to form the so-called *rainbow plane*, where each point corresponds

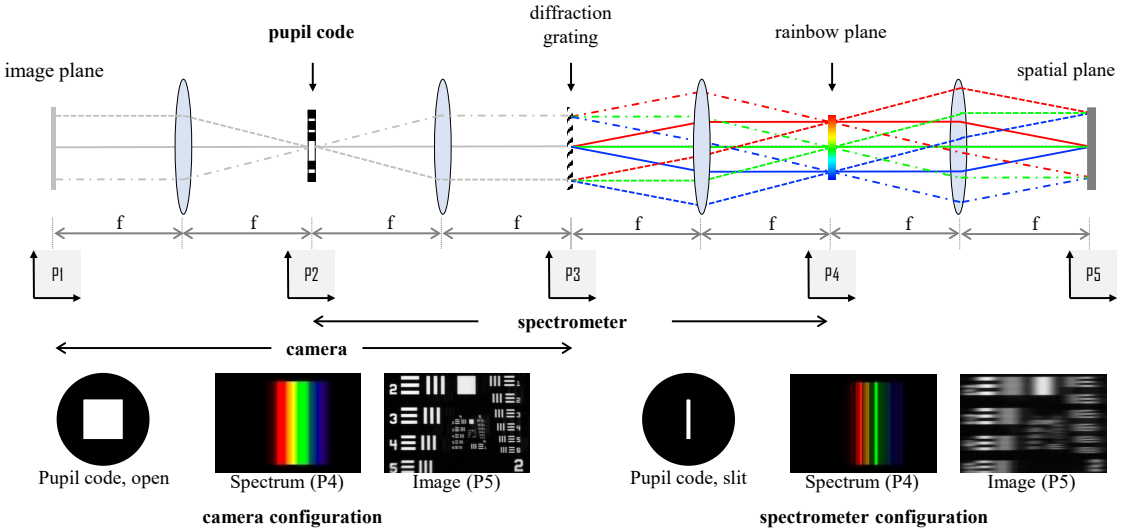


Figure 2.1: **Optical setup for capturing images with spectral programming**. P1 is the image plane of the objective lens, P2 contains spatial frequencies of the image, where we place a pupil code, $a(x, y)$. P3 contains the image plane blurred by the aperture function. We place a diffraction grating at this plane to disperse light into different wavelengths. P4 contains the resultant spectrum and P5 is a flipped copy of P3. In a camera configuration, the pupil code consists of an open aperture and leads to sharp image but blurred spectrum. In spectrometer configuration, the pupil code is a slit and leads to sharp spectrum but blurred image. This chapter formalizes the role played by pupil code for spatial and spectral resolutions.

to the average intensity of light of the whole scene for a single wavelength. The image on P3 is simply relayed on to plane P5. Arbitrary spectral programming can then be performed by placing an (SLM) on the rainbow plane (P4) and measuring image on plane P5. Intuitively, planes P1 to P3 is a simple camera with an aperture in its Fourier plane, and planes P2 to P4 is a spectrometer with the aperture replacing a slit. The setup provides an insight into the tradeoff between spatial and spectral resolutions. While a camera requires a large and open aperture for a compact spatial blur, this would lead to severe loss in spectral resolvability, as a spectrometer requires a narrow opening. Our goal is to formalize the role played by the shape of the pupil code in deciding spatial and spectral resolution.

2.1.2 Main result

We show that the pupil code $a(x, y)$ introduces a spectral and spatial blur, $h_\lambda(\lambda)$ and $h_x(x)$ respectively with standard deviations σ_λ and σ_x (detailed expression in eq. (2.20) and (2.21)). Our main contribution is in the form of a lower bound on the *space-spectrum bandwidth product* that relates the spectral resolution

at which light can be programmed and the spatial resolution of captured image. This is encapsulated in the following theorem.

Theorem 1 *For the spectrally-coded imaging architecture shown in Fig. 2.1, the product of spatial and spectral standard deviations σ_x and σ_λ, respectively, is bounded as*

$$\sigma_x \sigma_\lambda \geq \frac{\lambda}{4\pi v_0}, \tag{2.1}$$

where v_0 is the density of slits in the diffraction grating.

This result was first explored in [63] and [87] where the authors demonstrated that the spatial and spectral resolutions were related to the choice of pupil code. This chapter builds on their results by providing a concise expression for the tradeoff. We prove that a Gaussian-shaped pupil code achieves the lower bound and leads to most compact spatial blur for a targeted spectral blur.

2.1.3 Implications

The space-spectrum bandwidth product introduces an uncertainty in spectrally-programmable cameras, stating that one cannot arbitrarily program spectrum at high resolution without loss in spatial resolution. We demonstrate the impact of uncertainty by building a spectrally-programmable camera and showing that blocking one of two closely-spaced narrowband sources cannot be done without severe loss in spectral resolution. We also show that for narrowband filtering, the spatial blur is affected by the pupil code as well as the shape of the narrowband filter, and that a slit, a commonly used narrowband filter shape leads to a spectrally-varying spatial blur. Instead, using a Gaussian-shaped narrowband filter achieves *spectrally-independent* spatial blur, thereby being the optimal candidate for spectral programming.

Hyperspectral imagers. Apart from spectral programming, several hyperspectral imaging architectures [7, 58, 87] rely on obtaining spectrally-programmed images. Our findings impact such setups, as a key requirement of such setups is to capture high resolution images without sacrificing spectral resolution. Hence, the space-spectrum bandwidth product can serve as a design guide to carefully choose the pupil code to obtain desired spatial and spectral resolutions.

Spatially-coded cameras. We note that the analysis in this chapter is targeted specifically at spectrally-programmable cameras and does not apply to many hyperspectral cameras where there is no pupil plane coding. Cameras such as the pushbroom camera and the coded aperture snapshot spectral imager (CASSI) [49, 98] which scan the full HSI only perform spatial coding and, and as such, are not affected

by this result. Since such systems code space and then measure its spectrally-sheared image, the spatial code only affects the spectral resolution and not the spatial resolution.

2.2 Prior Work

We start our discussion by talking about capturing images with arbitrary spectral filters and then briefly state its applications. We then state the fundamental tradeoffs based on system parameters.

Measurement model. Consider a scene's hyperspectral image (HSI) represented by $H(x, y, \lambda)$, where (x, y) represent spatial coordinates and λ represents wavelength. Our goal is to optically obtain a spectrally-programmed image. Specifically, given a spectral filter $f(\lambda)$, our aim is to implement a camera that captures the following grayscale image,

$$I(x, y) = \int_{\lambda} H(x, y, \lambda) f(\lambda) d\lambda. \tag{2.2}$$

Applications of spectral programming. The ability to arbitrarily program spectrum enables a wide gamut of applications. This includes adaptive color displays [66], programmatically blocking illuminants [63], and detecting materials [88, 106]. The key advantage in all these applications is to not measure the complete HSI, but only the desired spectrally-programmed images; this leads to fewer measurements at higher signal to noise ratio (SNR). Such a system can also be used for compressively sensing the complete HSI [58, 87] which relies on capturing projection of a scene's HSI on random or designed spectral filters.

Spectrally-programmable camera architecture. Spectral programming is a technique that is often used in imaging applications, such as Bayer filters for RGB cameras or narrowband spectral filters for fluorescence microscopy [56]. Static filters offer arbitrarily high spectral resolution, but are not tailored for applications that require changing filters rapidly; while this can be achieved with filter wheels, the speed of such devices is constrained by the speed at which the filters can be changed. Electronically tunable filters, in part can be achieved by using a tunable filter [100] where liquid crystal (LC) cells are used to obtain a combination of narrowband spectral filters. LC filters however are typically slow as they require large settling times.

The most practical way of implementing programmable spectral filters that can be changed electronically and at high speeds is to rely on the setup shown in Fig 2.1. Here, a dispersion element such as a grating or prism is used to create the so-called *rainbow plane* [66] where each point corresponds to intensity of a single wavelength of the whole scene. By placing a spatial modulator (SLM) on this

plane, one can achieve arbitrary spectral programming. This approach is similar to replacing a sensor in a spectrometer with an SLM, and has been the *defacto* way of spectral programming in some of the past works [58, 63].

The SLM-based approach for spectral programming has certain advantages. Since SLMs are fast, one can achieve high frame rates (often in excess of 60fps), which is crucial for imaging dynamic scenes, and in applications that require rapidly switching spectral filters. Two, the system is *potentially* capable of high spectral resolution without sacrificing capture time. However, as we will see next, a high spectral resolution leads to a severe loss in spatial resolution. The focus of this paper is on the fundamental trade-off of spectral and spatial resolutions.

Time-frequency bandwidth product. Our main result is based on the time-frequency bandwidth product [22, 36, 77], which we state here for completeness. Let $x(t)$ be a centered time-domain signal, and let $X(\nu)$ be its (centered) continuous-time Fourier transform. We define the spread of time-domain and frequency-domain signals as,

$$\sigma_t = \frac{\sqrt{\int_t t^2 |x(t)|^2 dt}}{\sqrt{\int_t |x(t)|^2 dt}} \quad \text{and} \quad \sigma_\nu = \frac{\sqrt{\int_\nu \nu^2 |X(\nu)|^2 d\nu}}{\sqrt{\int_\nu |X(\nu)|^2 d\nu}}. \tag{2.3}$$

Then the uncertainty theorem states that,

$$\sigma_t \sigma_\nu \geq \frac{1}{4\pi}. \tag{2.4}$$

As a consequence, one cannot achieve simultaneous arbitrarily precise localization of time and frequency. The time-frequency bandwidth product finds application in various fields of signal processing, including optical systems [62, 104]. Our result is its translation to spatial and spectral signals which arises Fourier transform property of a thin lens[35].

Space-spectrum resolution tradeoff. In order understand the impact of the pupil code shape on spatial and spectral resolutions, let us consider the design of a spectrometer, which consists of a narrow opening, a dispersive element, and a sensing element. The spectral resolution of the measurements is a function of width of the opening slit; a narrower opening leads to high resolution, while a broad slit leads to blurred spectrum. Similarly, a programmable camera would also necessitate a narrow slit to ensure that spectrum can be modulated at high resolution. However, such a narrow slit leads to a severe loss in spatial resolution, since imaging at high resolution requires a large and open aperture. In [66], it is noted that a large slit leads to loss of spectral resolution, but they do not mention what happens to the spatial

resolution. The authors in [63] identified this tradeoff and stated an approximate relationship between spectral and spatial tradeoff for fully open aperture and demonstrated that high spectral resolution lead to blurry images. We formalize the result and show that such a tradeoff applies to any pupil code shape and can be concisely stated as a space-spectrum bandwidth product. In the upcoming sections, we will formalize the spatial and spectral resolutions that result from the choice of a pupil code.

2.3 Fundamental Limits of Spatial/Spectral Resolution

We now derive a concise lower bound on product of spatial and spectral spreads due to a spectrally-programmable camera.

Spectral and spatial blurs. Let us revisit the optical setup in Fig. 2.1, where we placed a pupil code $a(x)$ in plane P2 and obtained the rainbow plan on P4 and image on P5. We wish to study the effect of $a(x)$ on the blur it introduces in spectral and spatial measurements. For brevity, we show the blur along x-axis alone, as there is no spectral dispersion along y-axis. Let $a(x)$ be the shape of the aperture function and let $A(u)$ be its Fourier transform. Without loss of generality, we assume that $a(x)$ and $A(u)$ are both centered such that

$$\int x|a(x)|^2 dx = 0 = \int u|A(u)|^2 du. \tag{2.5}$$

We note that all our analysis is for spatially-incoherent light; any phase component is hence irrelevant. We rely on Fourier transform property of a thin lens [35] as well as the derivation in [87]. Assume that the complex field distribution on plane P1 is $i_1(x, y, \lambda)$. Then the field distribution on P2 that is $2f$ away is given by the scaled Fourier transform relationship, $i_2(x_2, y_2, \lambda) = \frac{1}{j\lambda f} I_1\left(\frac{x_2}{\lambda f}, \frac{y_2}{\lambda f}, \lambda\right)$, where $I_1(u, v)$ is the Fourier transform of $i_1(x, y)$. Propagating the signal through the optical setup simply requires us to perform such operations iteratively.

Consider a single spatial point on P1 of the form $i_1(x_1, y_1, \lambda) = s(x_0, y_0, \lambda)\delta(x_1 - x_0, y_1 - y_0)$, where $s(x_0, y_0, \lambda)$ is the complex amplitude of the point as a function of wavelength. Any arbitrary image can then be treated as infinite such point sources. The amplitude distribution on plane P2 is the scaled Fourier transform of amplitude on plane P1 and is given by,

$$i_2(x_2, y_2, \lambda) = \frac{1}{j\lambda f} s(x_0, y_0, \lambda) \exp\left\{-\frac{2\pi j}{\lambda f}(x_0 x_2 + y_0 y_2)\right\} \tag{2.6}$$

Let $a(x, y)$ be the complex amplitude of the pupil code placed on P2. Then the intensity just after the aperture is given by,

$$\widehat{i_2}(x_2, y_2, \lambda) = \frac{1}{j\lambda f} s(x_0, y_0, \lambda) \exp\left\{-\frac{2\pi j}{\lambda f}(x_0 x_2 + y_0 y_2)\right\} \times a(x_2, y_2) \tag{2.7}$$

With a similar derivation, we can show that the field distribution on P3 just before the diffraction grating is,

$$i_3(x_3, y_3, \lambda) = \frac{1}{j\lambda f}\widehat{I_2}\left(\frac{x_3}{\lambda f}, \frac{y_3}{\lambda f}\right) = \frac{1}{(j\lambda f)^2} s(x_0, y_0, \lambda) A\left(\frac{x_3 + x_0}{\lambda f}, \frac{y_3 + y_0}{\lambda f}\right), \tag{2.8}$$

where $A(u, v)$ is the Fourier transform of $a(x, y)$. For simplicity of analysis, we consider the diffraction grating to be a series of narrow band slits, modeled as an impulse train along x-axis as,

$$d(x, y) = \sum_{k=-\infty}^{\infty} \delta\left(x - \frac{k}{\nu_0}\right), \tag{2.9}$$

where ν_0 is the groove density. Then the propagated field just after the grating is given by,

$$\begin{aligned}\widehat{i_3}(x_3, y_3, \lambda) &= i_3(x_3, y_3, \lambda) d(x_3, y_3) \\ &= \frac{1}{(j\lambda f)^2} s(x_0, y_0, \lambda) A\left(\frac{x_3 + x_0}{\lambda f}, \frac{y_3 + y_0}{\lambda f}\right) \times \sum_{k=-\infty}^{\infty} \delta\left(x_3 - \frac{k}{\nu_0}\right)\end{aligned} \tag{2.10}$$

Using Fourier transform property of lens again, the field on P4 is,

$$\begin{aligned}i_4(x_4, y_4, \lambda) &= \frac{1}{j\lambda f}\widehat{I_3}\left(\frac{x_4}{\lambda f}, \frac{y_4}{\lambda f}\right) = \frac{1}{j\lambda f}\left(\frac{1}{j\lambda f}\right)^2 I_3\left(\frac{x_4}{\lambda f}, \frac{y_4}{\lambda f}\right) * D\left(\frac{x_4}{\lambda f}, \frac{y_4}{\lambda f}\right) \\ &= \frac{1}{j\lambda f}\left(\frac{1}{j\lambda f}\right)^2 I_3\left(\frac{x_4}{\lambda f}, \frac{y_4}{\lambda f}\right) * \left(\delta\left(\frac{y_4}{\lambda f}\right) \sum_{k=-\infty}^{\infty} \delta\left(\frac{x_4}{\lambda f} - k\nu_0\right)\right) \\ &= -\frac{1}{j\lambda f} s(x_0, y_0, \lambda) \sum_{k=-\infty}^{\infty} a(-(x_4 - k\nu_0\lambda f), -y_4) \exp\left\{j\frac{2\pi}{\lambda f}(x_0(x_4 - k\lambda f\nu_0) + y_0 y_4)\right\},\end{aligned} \tag{2.11}$$

leading to an equation which shows multiple, spectrally-dispersed copies of the aperture $a(x, y)$ along the x-axis. Our optical setup is designed to propagate only the first order and hence we retain the $k = 1$ copy, giving us,

$$i_4(x_4, y_4, \lambda) = -\frac{1}{j\lambda f} s(x_0, y_0, \lambda) \underbrace{a(-(x_4 - \nu_0\lambda f), -y_4)}_{\text{spectrally-shifted a(x, y)}} \exp\left\{j\frac{2\pi}{\lambda f}(x_0(x_4 - \lambda f\nu_0) + y_0 y_4)\right\}. \tag{2.12}$$

Finally, propagating the signal one more lens away, we get,

$$i_5(x, y, \lambda) = \frac{1}{(j\lambda f)^2} \exp\{-j2\pi x_5 \nu_0\}\, s(x_0, y_0, \lambda) A\left(-\frac{x_5 + x_0}{\lambda f}, -\frac{y_5 + y_0}{\lambda f}\right). \tag{2.13}$$

Intensity measurements. Consider cameras placed on planes P4 and P5 with a spectral response of $c(\lambda)$. The intensity measurement on P4,

$$\begin{aligned}M_4(x_4, y_4) &= \int_\lambda |i_4(x_4, y_4, \lambda)|^2 c(\lambda) d\lambda \\ &= \int_\lambda \frac{1}{\lambda^2 f^2} |s(x_0, y_0, \lambda)|^2 |a(-(x_4 - \nu_0\lambda f), -y_4)|^2 c(\lambda) d\lambda \\ &= \widehat{S}\left(x_0, y_0, \frac{x_4}{f\nu_0}\right) * |a(-x_4, -y_4)|^2,\end{aligned} \tag{2.14}$$

where $\widehat{S}(x_0, y_0, \lambda) = \frac{1}{\lambda^2 f^2}|s(x_0, y_0, \lambda)|^2 c(\lambda)$ is the measured intensity of the scene point. Extending to all points (x_0, y_0), we get,

$$M_4(x_4, y_4) = \int_{x_0}\int_{y_0} \widehat{S}\left(x_0, y_0, \frac{x_4}{f v_0}\right) * |a(-x_4, -y_4)|^2 = S\left(\frac{x_4}{\lambda f}\right) * |a(-x_4, -y_4)|^2. \tag{2.15}$$

Here, $S(\lambda)$ is an integral of spectrum of all spatial points. Equation (2.15) shows that the aperture function $a(x, y)$ results in spectral blur at every scene point. Similarly, the spatial image on P5,

$$\begin{aligned} M_5(x, y) &= \int_\lambda |i_5(x_5, y_5, \lambda)|^2 c(\lambda) d\lambda \\ &= \frac{1}{\lambda^4 f^4}\int_\lambda |s(x_0, y_0, \lambda)|^2 \left|A\left(-\frac{x_5 + x_0}{\lambda f}, -\frac{x_5 + x_0}{\lambda f}\right)\right|^2 c(\lambda) d\lambda. \end{aligned} \tag{2.16}$$

Computing intensity for all (x_0, y_0) gives us,

$$\begin{aligned} M_5(x, y) &= \int_{x_0}\int_{y_0} \frac{1}{\lambda^4 f^4}\int_\lambda |s(x_0, y_0, \lambda)|^2 \left|A\left(-\frac{x_5 + x_0}{\lambda f}, -\frac{x_5 + x_0}{\lambda f}\right)\right|^2 c(\lambda) d\lambda \\ &= \frac{1}{\lambda^4 f^4}\int_\lambda \underbrace{|s(x_5, y_5, \lambda)|^2 * \left|A\left(-\frac{x_5}{\lambda f}, -\frac{y_5}{\lambda f}\right)\right|^2}_{\text{Spatial blur}} c(\lambda) d\lambda. \end{aligned} \tag{2.17}$$

Equation (2.17) shows that the pupil code $a(x, y)$ introduces a spatial blur equal to a scaled version of its power spectral density (PSD), $|A(u, v)|^2$. For monochromatic light source, (2.17) is simply a convolution of scene's image with a scaled PSD of $a(x, y)$; for polychromatic sources, this expression has a spectrally-dependent PSF, which does not follow a convolution model. To make analysis simple, we assume that the shape of the PSF is approximately the same over a small range of wavelengths. Then the resultant expression for the spatial image is,

$$\begin{aligned} M_5(x_5, y_5) &= \left(\frac{1}{\lambda^4 f^4}|s(x_5, y_5, \lambda)|^2\right) * \left|A\left(-\frac{x_5}{\lambda_c f}, -\frac{y_5}{\lambda_c f}\right)\right|^2 \\ &= I(x_0, y_0) * \left|A\left(-\frac{x_5}{\lambda_c f}, -\frac{y_5}{\lambda_c f}\right)\right|^2, \end{aligned} \tag{2.18}$$

where $I(x_0, y_0)$ is a scaled grayscale image of the scene, and λ_c is a chosen, central wavelength of the spectral range.

Spectral and spatial blurs. For brevity and ease of understanding, we drop the y axis as it does not affect the spectral blur. From (2.15) and (2.18), we get the following expressions for spectral and spatial blurs,

$$h_\lambda(\lambda) = |a(-\lambda f v_0)|^2, \qquad h_x(x) = \left|A\left(-\frac{x}{\lambda f}\right)\right|^2. \tag{2.19}$$

We observe that the blur in space and spectrum are not independent; specifically, they form a Fourier-transform pair, with appropriate scaling. Our goal is to show that this interdependence between spatial and spectral blur has a very specific structure and their product can be lower bounded – implying that we cannot arbitrarily resolve in both domains.

2.3.1 The space-spectrum uncertainty principle

Our main result, stated in Theorem 1, suggests that the spatial and standard deviations are related by the inequality, $\sigma_x \sigma_\lambda \geq \frac{\lambda}{4\pi v_0}$. We now outline the proof of our theorem.

Proof. The spectral and spatial standard deviations are,

$$\sigma_\lambda = \sqrt{\frac{\int_\lambda \lambda^2 h_\lambda(\lambda) d\lambda}{\int_\lambda h_\lambda(\lambda) d\lambda}} = \sqrt{\frac{\int_\lambda \lambda^2 |a(-\lambda f v_0)|^2 d\lambda}{\int_\lambda |a(-\lambda f v_0)|^2 d\lambda}} \tag{2.20}$$

$$\sigma_x = \sqrt{\frac{\int_x x^2 h_x(x) dx}{\int_x h_x(x) dx}} = \sqrt{\frac{\int_x x^2 \left|A\left(-\frac{x}{\lambda f}\right)\right|^2 dx}{\int_x \left|A\left(-\frac{x}{\lambda f}\right)\right|^2 dx}}, \tag{2.21}$$

which are similar to time and frequency spreads defined in eq. (2.3) with appropriate scaling. Given that σ_t is the spread of $x(t)$, the spread of a scaled function $\widehat{x}(t) = x(st)$ is $\widehat{\sigma}_t = s\sigma_t$. From eq. (2.4) and substituting $t = \lambda f v_0$ and $\nu = \frac{x}{f\lambda}$,

$$\left(\frac{1}{f^2\lambda^2}\right) \sigma_x^2 (f^2 v_0^2) \sigma_\lambda^2 \geq \frac{1}{16\pi^2} \implies \sigma_x^2 \sigma_\lambda^2 \geq \frac{\lambda^2}{16\pi^2 v_0^2}$$

$$\boxed{\sigma_x \sigma_\lambda \geq \frac{\lambda}{4\pi v_0}} \tag{2.22}$$

Implication. We make some observations about the uncertainty principle here.

- *Invariance to scaling.* The bandwidth product does not change even if the aperture is stretched or squeezed. If the aperture $a(x)$ is replaced by $a(sx)$, then spectral blur changes to $h_\lambda(\lambda) = |a(-s\lambda f v_0)|^2$ and the spatial blur changes to $\left|A\left(-\frac{x}{s\lambda f}\right)\right|^2$. This changes the spectral and spatial variances to $s^2\sigma_\lambda^2$ and σ_x^2/s^2, thereby keeping the product a constant.

- *Invariance to power of lenses.* The bandwidth product is independent of focal length of the system, implying that one cannot expect any increase in product of standard deviations by changing the lenses.

- *Dependence on groove density.* The bandwidth product inversely depends on the groove density v_0. In theory, one can achieve arbitrarily low space-bandwidth product by having high groove density, but the limiting factor becomes the aperture size of lenses.

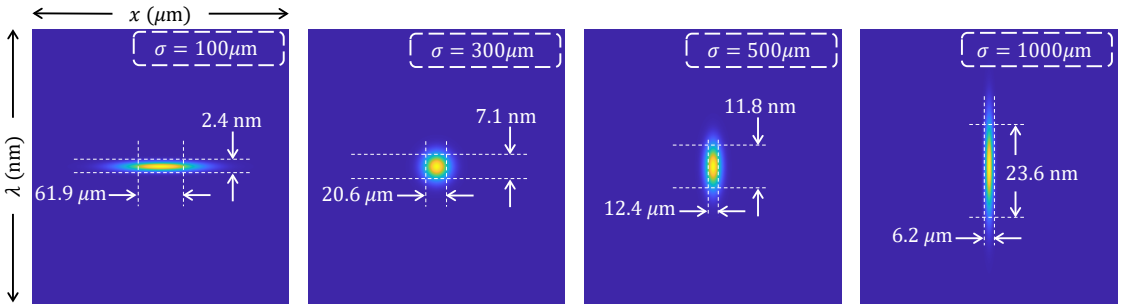


Figure 2.2: **Simulated x – λ blur for Gaussian window**. The four figures illustrate the spatio-spectral blur for various window sizes. The blur kernel was computed for $\lambda = 500$ nm, $f = 100$ mm and a groove density of 300 grooves/mm. There is a visible trade off between the two resolutions. The appropriate window size depends on the application; a camera with low spectral resolution requirement can use a $\sigma = 500\mu$m window, while one with stringent spatial resolution requirements may use a $\sigma = 1000\mu$m window.

- *Dependence on wavelength.* The bandwidth product is directly proportional to wavelength. This is expected, as the limiting case of our statement, where σ_λ is several hundreds of nanometers is just a normal grayscale imager, and in that case, the expression looks very similar to Abbe's diffraction limit [59]. However, one may make the expression independent of wavelength by using lower bound of the spectral range,

$$\boxed{\sigma_x \sigma_\lambda \geq \frac{\lambda_{\min}}{4\pi v_0}} \tag{2.23}$$

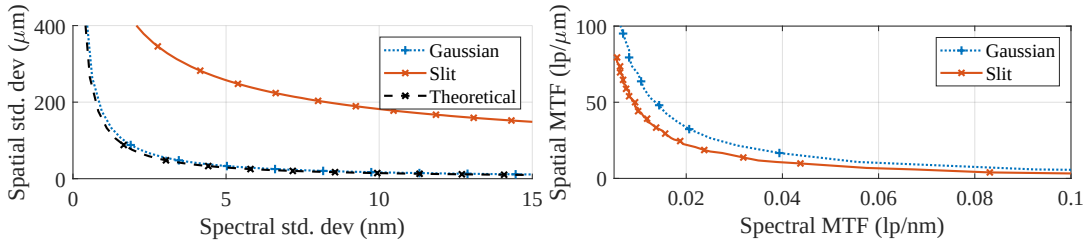
Achievability of lower bound. As in the case of time-frequency uncertainty, there exists a pupil code function that has its space-spectrum bandwidth product *equal* to $\frac{\lambda}{4\pi v_0}$. This is achieved by the family of Gaussian windows:

$$a(x, y) = \exp\left\{-\frac{x^2}{2\sigma^2}\right\}. \tag{2.24}$$

The spectral and spatial blur are then given by,

$$\widetilde{f}(\lambda) = \exp\left\{-\frac{\lambda^2 f^2 v_0^2}{\sigma^2}\right\}, \qquad \widetilde{g}(x) = \exp\left\{-\frac{4\pi^2\sigma^2 x^2}{\lambda^2 f^2}\right\}. \tag{2.25}$$

Figure 2.2 shows the simulated "uncertainty" box at 500 nm of Gaussian windows of various widths. We simulated a system comprising of 100 mm lenses and a diffraction grating with a groove density of 300 grooves/mm. Evidently, as we squeeze along one axis, the other axis stretches with the product of widths being a constant at 145.9 nm·μm.



(a) Standard deviation

(b) MTF at 30% contrast ratio

Figure 2.3: **Simulations on common aperture shapes**. (a) compares spectral and spatial standard deviations whereas (b) shows spectral and spatial MTF at 30% contrast. Gaussian codes achieve theoretical limit when resolution metric is standard deviation of window.

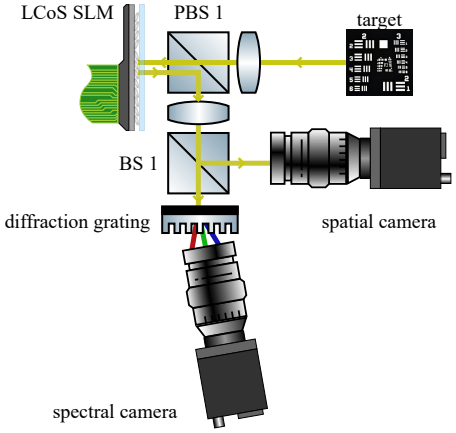
2.3.2 Verification using simulations

We provide a validation of our theory with simulations. We specifically compared a box aperture that simulates a slit or fully open aperture, and a Gaussian aperture. For the purpose of exposition, we used $f = 75$ mm and a diffraction grating of 300 groves/mm. Figure 2.3(a) shows a plot of spatial and spectral standard deviations and (b) shows a plot of spatial and spectral modulation transfer function (MTF) at 30% contrast ratio. The plots show a clear trade off between the two resolutions, independent of resolution metric. We also observe that Gaussian codes achieve the theoretical limit for standard deviation. Hence we conclude that the space-spectrum bandwidth product is a tight bound. Next, we validate our findings with an optical setup that implements the schematic in Fig. 2.1 and capture scenes with various aperture shapes.

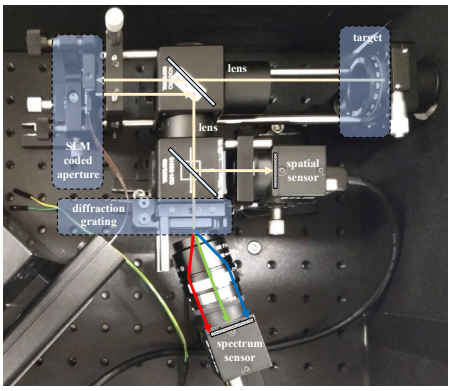
2.4 Experiments

Armed with our theoretical insights, we next verify the results with some real experiments.

Optical setup. We built an optical setup shown in Fig. 2.4 with relevant components marked. The setup is a minor modification of the schematic shown in Fig. 2.1. We placed a spatial light modulator (SLM) on plane P2 which enabled display of various coded apertures, and a diffraction grating in plane P3. The spectral measurement camera is on plane P4. Instead of placing spatial camera on P5, we place it on P3 (using beamsplitter BS1). Since we do not code the rainbow plane P4, image on P3 and P5 are equivalent. Focal length of all our lenses was 75 mm and the diffraction grating had 300 grooves/mm. List of components can be found in appendix B.



(a) Schematic



(b) Lab prototype

Figure 2.4: **Schematic and image of our lab prototype**. We displayed various patterns on SLM to form coded apertures to evaluate spatial and spectral resolutions. The spectral camera was tilted to capture the first order of diffraction from the grating.

Visualization of spectral and spatial resolutions. To illustrate our hypothesis, we placed a USAF resolution chart on the image plane P1. The scene was illuminated with a cool white compact fluorescent lamp (CFL) which is comprised of several narrow peaks. This setup enabled us to simultaneously visualize sharp spectrum as well as sharp spatial features. Figure 2.5 shows images and spectra for some representative cases. Each row shows results for a specific coded aperture, whereas each column shows results for a fixed spectral resolution. The trend of decreasing spectral resolution with increasing spatial resolution is clearly visible. Further, a Gaussian aperture is superior to slit in terms of greater spatial resolution for the same spectral resolution, which agrees with our theoretical findings.

Quantitative verification. We illuminated a pinhole with a spectrally-narrowband light source with a central wavelength of 670 nm and an FWHM of 3 nm. We then captured both spectrum of the light source and image of the pinhole, which we then used for computing the corresponding standard deviations. Figure 2.6 compares reciprocal of spatial resolution against spectral resolution. The two plots show a straight line, thereby verifying that the product of spatial and spectral resolutions is a constant. We also observe that the line for Gaussian aperture is very close to the theoretically optimal line, thereby confirming that the lower bound is tight, even in practice.

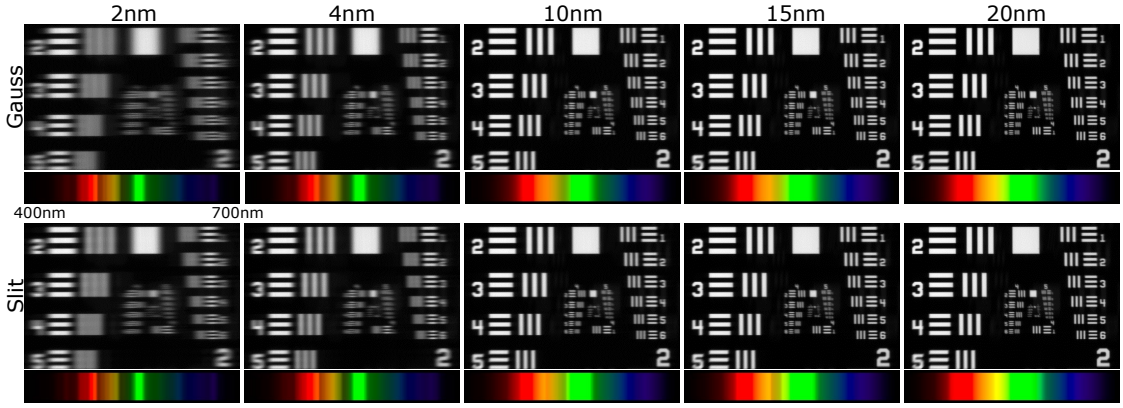
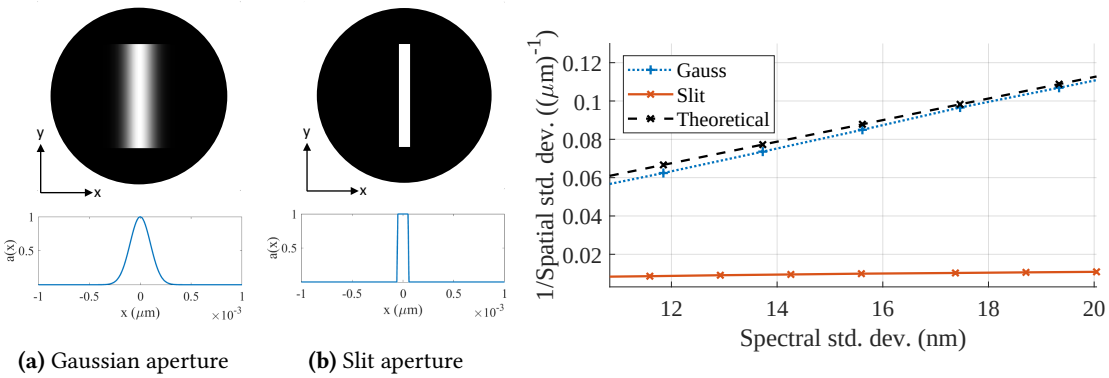


Figure 2.5: **Visualization of spectral and spatial resolutions**. We illuminated a USAF resolution target with a CFL lamp. We varied aperture types and widths to get spatial and spectral measurements. Each row shows image and spectrum for a specific aperture, and each column shows image and spectrum for a fixed spectral standard deviation. The results clearly illustrate the tradeoff between spatial and spectral resolution.



(a) Gaussian aperture **(b)** Slit aperture

Figure 2.6: **Quantitative measurement of resolutions**. We captured spatial and spectral measurements using the setup in Fig. 2.4. We illuminated a pinhole with a narrowband light source at 670 nm and swept across various sizes of (a) Gaussian and (b) slit apertures. We plot the reciprocal of spatial standard deviation plotted against spectral standard deviation, clearly showing a straight line.

2.4.1 Spectral programming

Next, we discuss the impact of our findings for various scenarios of spectral programming.

Effect of edge-pass filter. The tradeoff between spectral and spatial resolution affects how well the spectrum can be coded. To test this, we illuminate two closely spaced spatial points in a scene with two narrowband light sources (520 nm and 532 nm). We then attempt to block the 520 nm laser with various

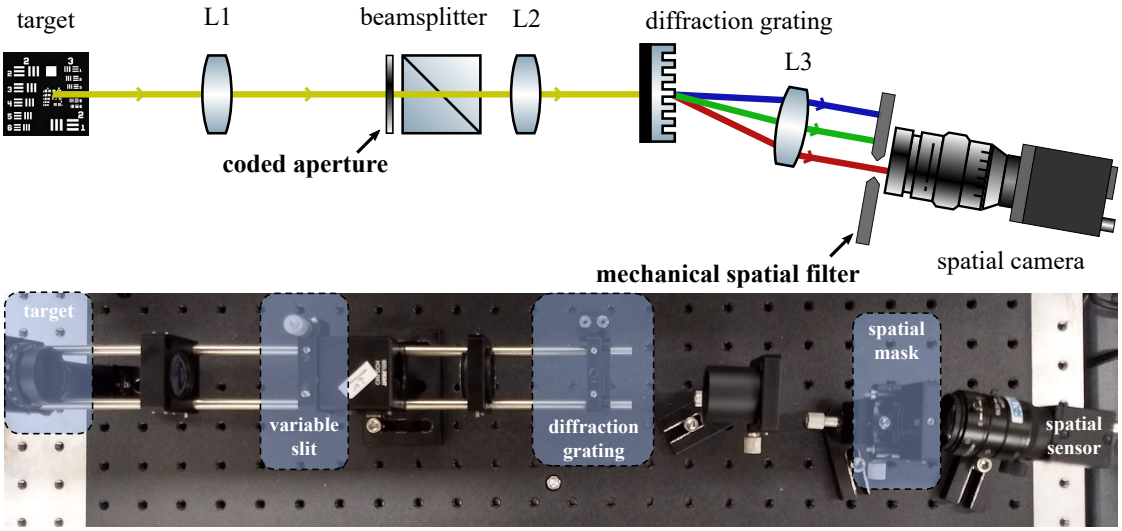


Figure 2.7: **Schematic and image of our prototype for spectral programming**. We illuminated a USAF target with 520 nm and 532 nm lasers. We then blocked 532 nm with a spatial mask on the rainbow plane. Images were captured with various slit widths to show the effect of space-spectrum uncertainty.

coded apertures and observe the spatial image. Figure 2.7 shows the schematic and our lab prototype for spectral programming, which is similar to the schematic shown in Fig. 2.1 with a spatial mask placed in plane P4. The results are shown in Fig. 2.8. With a broad aperture, it is not possible to effectively block one of the two lasers, shown in fourth column. A narrow aperture can lead to effective blocking (compare first and last columns) but with loss in resolution.

Effect of the shape of a narrowband filter. We now show that a slit has unintended implication, when used as a narrow band filter. In order to perform narrowband spectral programming, an intuitive choice is to place a narrow slit on the rainbow plane P4 and a camera on plane P5. This results in a spectra that looks similar to the example in Fig. 2.9(a). While such a mask works well for the target wavelength, the spatial images corresponding to adjacent wavelengths have severe loss in spatial resolution.

To understand the effect of a narrowband filter, consider a scene illuminated by a monochromatic light source of wavelength λ_1. The resulting field on rainbow plane,

$$i_4(x) = a\left(x - \lambda_1 f \nu_0\right). \tag{2.26}$$

Now let a spatial mask $\widehat{a}(x)$ centered around λ_2 be placed on the rainbow plane. Then the output just

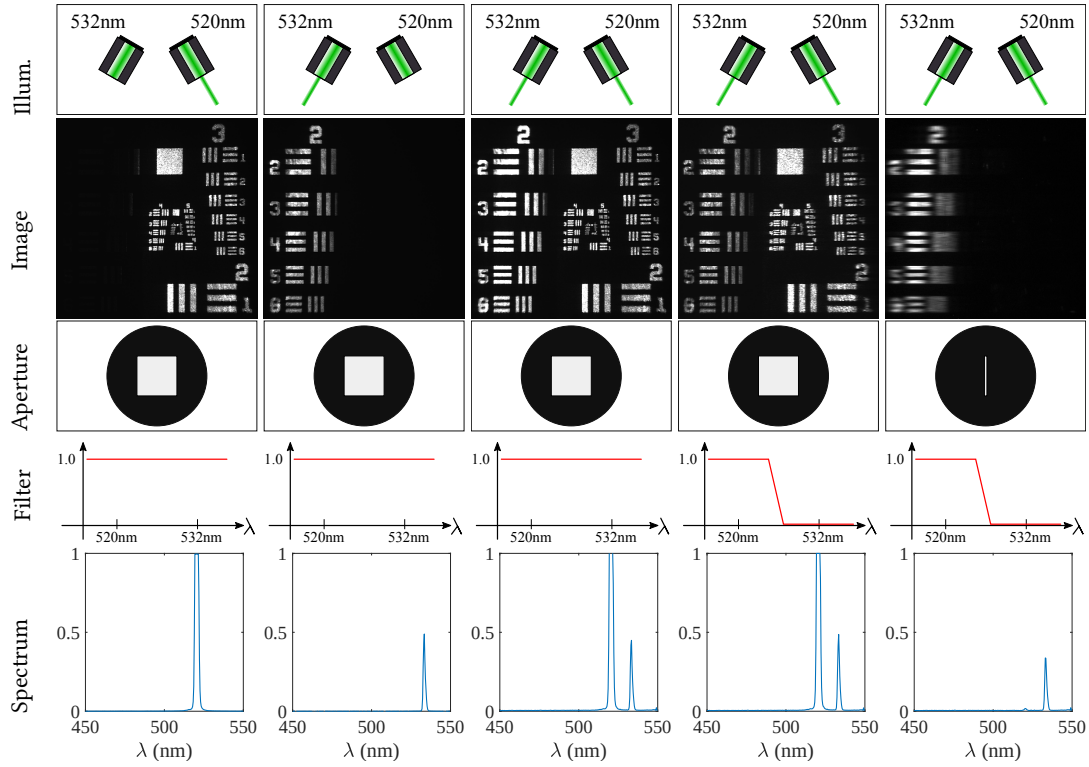


Figure 2.8: **Spectral programming with narrowband sources**. A consequence of space-spectrum bandwidth product is the incapability of spectral programming at high resolution. In this example, we show how blocking one of the two closely spaced narrowband lasers and only be done with severe loss in resolution.

after the mask,

$$\widetilde{i}_4(x) = a\,(x - \lambda_1 f \nu_0)\,\widehat{a}\,(x - \lambda_2 f \nu_0)\,. \tag{2.27}$$

If both $a(x)$ and $\widehat{a}(x)$ are slits of width W, then the effective width is $W - |\lambda_1 - \lambda_2|$, which decreases with increasing gap between the two wavelengths. This is illustrated by the plots of PSFs in Fig. 2.9(c). We utilized the setup in Fig. 2.7, where we illuminated a pinhole with a 520 nm laser. We placed a spatial mask on a horizontal translation stage to block various adjacent wavelengths. We then measured the image of a pinhole and fit an appropriate curve to the measured PSF. Evidently, the PSF has a larger spread as the gap between target wavelength and central wavelength of filter increases. This is true even if the pupil code were Gaussian mask and the filter were a slit, as shown in Fig. 2.9(d). Ideally we require the effective width to be independent of λ_1 and λ_2. This is achieved if both pupil plane and the

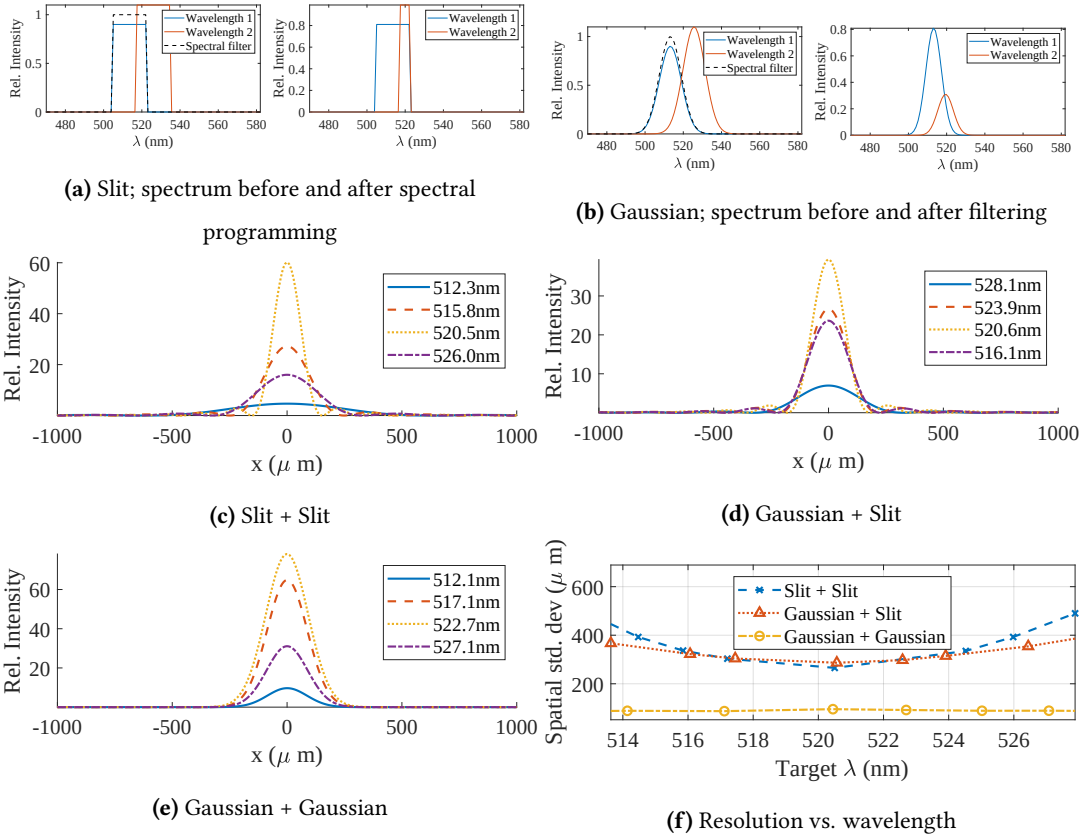


Figure 2.9: **Effect of narrowband spectral programming on spatial resolution**. Spatial resolution is affected by the pupil code as well as the spatial mask used for creating a narrowband spectral filter. We illuminated a pinhole with a 520 nm laser and then swept various spatial masks to filter wavelengths around 520 nm. For configurations where either the pupil code or the spatial mask was a slit, the spatial resolution got worse with increasing gap between desired and laser wavelength. In contrast, a configuration with both masks being Gaussian resulted in a wavelength-independent spatial blur.

filter have a Gaussian shape. In such a case the field,

$$\begin{aligned}\widetilde{i}_4(x) &= \exp\left\{-(x-\lambda_1 f\nu_0)^2/\sigma^2\right\}\exp\left\{-(x-\lambda_2 f\nu_0)^2/\sigma^2\right\}\\ &= \underbrace{\exp\left\{-(\lambda_1-\lambda_2)^2 f^2\nu_0^2/\sigma^2\right\}}_{\text{amplitude}}\underbrace{\exp\left\{-2(x-(\lambda_1+\lambda_2)f\nu_0/2)^2/\sigma^2\right\}}_{\text{aperture shape}}.\end{aligned} \tag{2.28}$$

The output field has a spread that is independent of λ_1, λ_2 which results in a wavelength-independent PSF. This is illustrated by the plot of PSFs in 2.9(e) with a gaussian aperture as well as filter shape. Figure 2.9(f) compares PSF spread in terms of spatial standard deviation for various positions of filter, clearly

illustrating the wavelength-indepedent blur arising due to Gaussian-shaped pupil-code and filter.

2.5 Conclusion

We formalized the tradeoff between spectral and spatial resolution associated with a spectrally--programmable camera of the type shown in Fig. 2.1 and stated the space-spectrum uncertainty principle. We showed through theory, simulations and real experiments that one can finely resolve space or spectrum, but not both. Our analysis then showed that a Gaussian-shaped aperture achieves the theoretical lower bound, and that a Gaussian-shaped narrowband filter introduces a wavelength-independent spatial blur. We believe our findings impacts scientific imaging at large by providing insights and design guidelines for settings which rely on spectral programming.

Application to fundamental limits of hyperspectral imaging. We note that our analysis *does not* limit the spatial and spectral resolution of hyperspectral cameras. In case of cameras which utilize tunable filters, the spatial resolution is a function of the grayscale camera, while the spectral resolution is *independently* dictated by the tunable filter. For spatially-coded cameras such as pushbroom and CASSI, the sensing process does not rely on pupil coding, which does not produce diffractive blur. To understand this, consider a spatially coded camera that has only one opening in the spatial plane, $\delta(x - x_0, y - y_0)$. The measurement on the camera sensor after propagating through a dispersive element is,

$$I_{\text{cam}}(x, y) = \int_{\lambda} \delta(x - x_0 - \Delta(\lambda), y_0), \tag{2.29}$$

where $\Delta(\lambda)$ is a wavelength-dependent shift. We observe that the image is simply a spectrally-smeared version of the spatial point, with no extra blur (other than the one caused by aberrations due to optics). In a sense, if the desired application requires a full scan of the scene's HSI, then it is possible to achieve high spatial and spectral resolution. However, the analysis in this chapter targets spectrally-programmable cameras. As we saw in the prior work section, such cameras are indispensable to efficiently sense [87] and infer [63, 88] HSIs. A practical and fast implementation of spectrally-programmable cameras requires pupil coding, which limits the simultaneously achievable spatial and spectral resolutions.

3 KRISM – Krylov Subspace-based Optical Computing of Hyperspectral Images

With a good insight into building spectrally-programmable cameras, we now turn our attention to an efficient way of capturing HSI of a scene. We will see that the dominant singular vectors of a scene's HSI can be sensed by alternating between spectrally-programmable imaging and spatially-coded spectrometry, by feeding the output of one stage to another. Toward the end of this chapter, we will learn about using computational tools to overcome the space-spectrum uncertainty product, methods to optically compute the singular value decomposition of a scene's HSI, and build a compact and efficient optical system to achieve this decomposition.

3.1 Introduction

Hyperspectral images (HSIs) capture light intensity of a scene as a function of space and wavelength and have been used in numerous vision [48, 72, 95], geo-science and remote sensing applications [19, 39]. Traditional approaches for hyperspectral imaging, including tunable spectral filters and pushbroom cameras, rely on sampling the HSI, i.e., measuring the photon counts in each spatio-spectral voxel. When imaging at high-spatial and spectral resolutions, the amount of light in a voxel can be quite small, thus requiring long exposures to mitigate the effect of noise.

HSIs are often endowed with rich structures that can be used to alleviate the challenges faced by traditional imagers. For example, natural scenes are often comprised of a few materials of distinct spectra and further, illumination of limited spectral complexity [52, 74]. This implies that the collection of spectral signatures observed at various locations in a scene lies close to a low-dimensional subspace. Instead of sampling the HSI of the scene one spatio-spectral voxel at a time, we can dramatically speed-up acquisition and increase light throughput by measuring only projections on this low-dimensional subspace. However, such a measurement scheme requires a priori knowledge of the scene since this subspace is entirely scene dependent. This chapter introduces an optical computing technique that identifies this

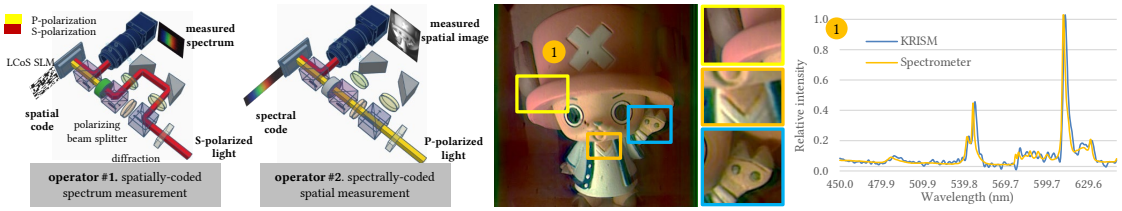


Figure 3.1: **Optical Computing of HSIs.** Hyperspectral imagers resolve scenes at high spatial and spectral resolutions. We propose a novel architecture called KRISM that provides the ability to implement two operators: a spatially-coded spectrometer and a spectrally-coded spatial imager. By iterating between the two, we can acquire a low rank approximation of the hyperspectral image in a light efficient manner with very few measurements. The left image shows optical schematics for implementing the two operators. On the right, we show a hyperspectral image of a scene illuminated with a compact fluorescent lamp (CFL) acquired using our lab prototype. The proposed method enables high spatial and spectral resolution as observed in the zoomed-in image patches and CFL peaks, respectively.

subspace using an iterative and adaptive sensing strategy and constructs a low-rank approximation to the scene's HSI. The proposed imager senses a low-rank approximation of a HSI by optically implementing the so-called Krylov subspace method [34]. We show that this requires two operators: a spatially-coded spectrometer and a spectrally-coded spatial imager; when we interpret the HSI as a 2D matrix, these two operators correspond to left and right multiplication of the matrix with a vector. The two operators are subsequently used in an iterative and adaptive imaging procedure whose eventual output is a low-rank approximation to the HSI. The proposed imager is adaptive, i.e., the measurement operator used to probe the scene's HSI at a given iteration depends on previously made measurements. This is a marked departure from current hyperspectral imaging strategies where the signal model is merely used as a prior for recovery from non-adaptive measurements [3].

Contributions. We propose an optical architecture that we refer to as KRylov subspace-based Imaging and SpectroMetry (KRISM) and make the following three contributions:

- *Optical computation of HSIs.* We show that optical computing of HSIs to estimate its dominant singular vectors provides significant advantages in terms of increased light throughput and reduced measurement time.

- *Coded apertures for resolving space and spectrum.* Sensing architectures typically used in spectrometry and imaging are mutually incompatible due to use the of slits in spectral imaging and open apertures in conventional imaging. To mitigate this, we study the effect of pupil plane coding on the HSI and

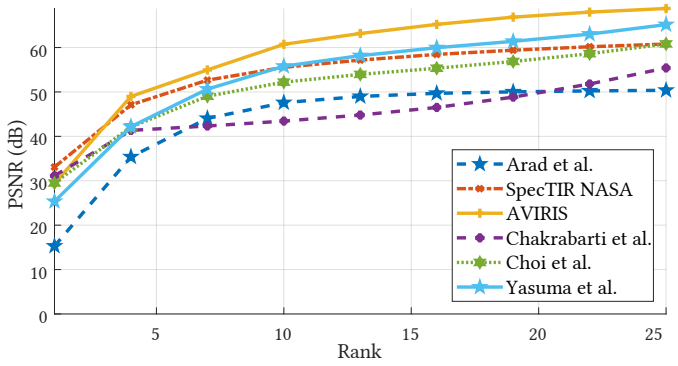


Figure 3.2: **Concise representation of HSIs with a low-rank model**. HSIs, interpreted as a matrix, are often low rank. We validate this observation by plotting accuracy in terms of peak SNR (PSNR) as a function of the rank of the approximation. We do this for many commonly used HSI datasets and observe that the PSNR is higher than 40dB for a rank 10 approximation across all datasets.

propose a coded aperture design that is capable of simultaneously achieving high spatial and spectral resolutions.

- *Optical setup.* We design and validate a novel and versatile optical implementation for KRISM that uses a single camera and a single spatial light modulator (SLM) to efficiently implement spatially-coded spectral and spectrally-coded spatial measurements.

The contributions above are supported via an extensive set of simulations as well as real experiments performed using the lab prototype.

Limitation. The benefits and contributions described above come with a key limitation. Our method is only advantageous if there are a sufficient number of spectral bands and the hyperspectral image is sufficiently low rank. If we only seek to image with very few spectral bands or if the scene is not well approximated by a low-rank model, then the proposed method performs poorly against traditional sensing methods.

3.2 Prior work

Nyquist sampling of HSIs. Classical designs for hyperspectral imaging based on Nyquist sampling include the tunable filter — which scans one narrow spectral band at a time, measuring the image associated with spectral bands at each instant — or using a pushbroom camera — which scans one spatial row at a time, measuring the entire spectrum associated with each pixel on the row. Both approaches

Method	Approach	Number of measurements	Estimation accuracy under noise	Advantages	Disadvantages
Sampling	Tunable spectral filter	$N_x N_y N_\lambda$	$\sigma\sqrt{N_x N_y N_\lambda}$	Easy calibration	Low spectral resolution; high acquisition time
	Pushbroom			High spectral resolution	Optical complexity; high acquisition time
Multiplexed	Spatial multiplexing	$N_x N_y N_\lambda$	$\sigma\sqrt{N_\lambda}$	Hadamard multiplexing gain	High acquisition time
	Spectral multiplexing		$\sigma\sqrt{N_x N_y}$		
Compressive sensing	CASSI	depends on signal model		Fewer measurements	Loss in spatial/spectral resolution
	Row/column projection	$\propto k^2(N_x N_y + N_\lambda)$	[Fazel et al. 2008]		Complex optics
KRISM (proposed method)	Optical Krylov subspace	$\propto k(N_x N_y + N_\lambda)$	prop. to model misfit + noise	Fewest number of measurements; very high light efficiency	Complex optics

Table 3.1: **Comparison of HSI sensing strategies**. Various sensing strategies for hyperspectral imaging of $N_x \times N_y$ spatial dimension and N_λ spectral bands. Noise in measurement is assumed to be AWGN with σ^2 variance. The expressions in third column represent the number of measurements required, while those in fourth column represent the error in reconstruction.

are time-consuming as well as light inefficient since each captured image wastes a large percentage of light incident on the camera.

Multiplexed sensing. The problem of reduced light throughput can be mitigated by the use of multiplexing. One of the seminal results in computational imaging is that the use of multiplexing codes including the Hadamard transform can often lead to significant efficiencies either in terms of increased SNR or faster acquisition [40]. This can either be spectral multiplexing [66] or spatial multiplexing [93]. While multiplexing mitigates light throughput issues, it does not reduce the number of measurements required. Sensing at high spatial and/or spectral resolution still requires long acquisition times to maintain a high SNR. Fortunately, HSIs have concise signal models that can be exploited to reduce the number of measurements.

Low-rank models for HSIs. There are many approaches to approximate HSIs using low-dimensional models; this includes group sparsity in transform domain [83], low rank model [33, 54], as well as low-rank and sparse model [89, 99]. Of particular interest to us is the low-rank modeling of HSIs when they are represented as a 2D matrix (See Figure 3.2). These models have found numerous uses in vision and graphics including color constancy [28], color displays [47], endmember detection [101], source separation [45], anomaly detection [89], compressive imaging [33] and denoising [105]. Chakrabarti and Zickler [15] also provide empirical justification that HSIs of natural scenes are well represented by low dimensional models.

Compressive hyperspectral imaging. The low-rank model has also been used for compressive sensing (CS) of HSIs. CS aims to recover a signal from a set of linear measurements that are fewer than its dimensionality [9]. This is achieved by modeling the sensed signal using lower dimensional representations — low-rank matrices being one such example. The technique most relevant to this chapter is that of row/column projection [27] where the measurement model is restricted to obtaining row and column projections of a matrix. Given a matrix $X \in \mathbb{R}^{m \times n}$, and measurement operators $S_{\text{row}} \in \mathbb{R}^{p \times m}, S_{\text{column}} \in \mathbb{R}^{n \times p}$, the measurements acquired are of the following form,

$$Y_{\text{row}} = S_{\text{row}} X, \quad Y_{\text{column}} = X S_{\text{column}}.$$

When the matrix X has a rank k, it can be shown that it is sufficient to acquire p images and p spectral profiles with $p \propto k^2$. In contrast, the proposed method requires only a number of measurements proportional to the rank of the matrix; however, these measurements are adaptive to the scene. At an increased cost of optical complexity, adaptive sensing promises accurate results with fewer measurements than non-adaptive measurement strategies.

Hyperspectral imaging architectures. Several architectures have been proposed for CS acquisition of HSIs. The Dual-Disperser Coded Aperture Snapshot Spectral Imager (DD-CASSI) [31] obtains a single image multiplexed in both spatial and spectral domains by dispersing the image with a prism, passing it through a coded aperture, and then recombining with a second prism. In contrast, the Single Disperser CASSI (SD-CASSI) [98] relies on a single prism that performs spatial coding using a binary mask followed by spectral dispersion with a prism. Baek et al. [8] disperse the image by placing a prism right before an SLR camera. The HSI is then reconstructed by studying the dispersion of color at the edges in the obtained RGB image. Takatani et al. [94] instead propose a snapshot imager that uses a faced reflectors overlaid with color filters. Various other snapshot techniques have been proposed which rely on space-spectrum multiplexing [13, 46, 57]. While snapshot imagers require only a single image, they often produce HSIs with reduced spatial or spectral resolutions. Data-driven approaches such as overcomplete dictionaries [57] and convolutional neural networks [18] partially alleviate the loss in resolution by building priors for the HSI. However, they require complex optimization that can often be time consuming.

Resolution and accuracy of the HSI can be improved by acquiring multiple measurements instead of a single snapshot image. Examples include multiple spatio-spectrally encoded images [49], spatially-multiplexed spectral measurements [54, 93] or separate spatial and spectral coding [58]. While multi-

measurement techniques overcome spatial and spectral resolution limits, the price is paid in the form of increased number of measurements and hence, reduced time resolution.

Performance of snapshot techniques can be improved by tailoring the spatial masks to a given HSI dataset [84, 85] or by optimizing spatial masks for sensing a selected subset of spectral bands [4]. Optimizing the spatial masks results in increased accuracy, but still requires long reconstruction times. A key insight into the existing methods is that the measurements are either non-adaptive and random, or adapted to a fixed signal class. In contrast, the proposed method is *adapted to the specific instance of the signal*, requires fewer measurements, and has practically no post-processing for reconstruction. Table 3.1 compares and contrasts various HS imaging strategies and their relative merits in terms of number of measurements and error in reconstruction. We next discuss the concept of Krylov subspaces for low-rank approximation of matrices, which motivates iterative and adaptive techniques and paves the way to the proposed method.

Krylov subspaces. Central to the proposed method is a class of techniques, collectively referred to as Krylov subpaces, for estimating singular vectors of matrices. Recall that the singular value decomposition (SVD) of a matrix $X \in \mathbb{R}^{m\times n}, m \leq n$ is given as $X = U\Sigma V^\top$, where $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ are orthonormal matrices, referred to as the singular vectors, and $\Sigma \in \mathbb{R}^{m\times n}$ is a diagonal matrix of singular values. Krylov subspace methods allow for efficient estimation of the singular values and vectors of a matrix and enjoy two key properties. First, we only need access to the matrix X via left and right multiplications with vectors, i.e., we do not need explicit access to the elements of the matrix X. Second, the top singular values and vectors of a low-rank matrix can be estimated using a small set of matrix-vector multiplications. These two properties are invaluable when the matrix is very large or when it is implicitly represented using operators or, as is the case in this chapter, the matrix is the scene's HSI and we only have access to optical implementations of the underlying matrix-vector multiplications.

There are many variants of Krylov subspace techniques which differ mainly on their robustness to noise and model mismatch. The techniques in this chapter are based on the so-called Lanczos bidiagonalization with full orthogonalization [34, 43]. Such iterative operations to reduce the complexity of matrix-vector multiplications have found use in communication theory in the form of reduced-rank filtering [30, 96] and adaptive beam forming [29]. Our goal is to leverage the benefits of iterative operations for low-rank approximation of high dimensional optical signals, in particular HSIs.

Optical computing of low-rank signals. Matrix-vector and matrix-matrix multiplications can often be implemented as optical systems. Such systems have been used for matrix-matrix multiplication [6],

matrix inversion [78], as well as computing eigenvectors [50]. Of particular interest to our method is the optical computing of the light transport operator using Krylov subspace methods [70]. The light transport matrix T represents the linear mapping between scene illumination and a camera observing the scene. Each column of the matrix T is the image of the scene when only a single illuminant is turned on. Hence, given a vector $\boldsymbol{\ell}$ that encodes the scene illumination, the image captured by the camera is given as $\mathbf{r} = T\boldsymbol{\ell}$. By Helmholtz reciprocity, if we replaced every pixel of the camera by a light source and every illuminant with a camera pixel, then the light transport associated with the reversed illumination/sensing setup is given as $T^\top$. Hence, by co-locating a projector with the camera and a camera with the scene's illuminants, we have access to both left- and right-multiplication of the light transport matrix with vectors; we can now apply Krylov subspace techniques for *optically* estimating a low-rank approximation to the light transport matrix. This delightful insight is one of the key results in [70].

We propose a translation of the ideas in [70] to hyperspectral imaging. However, as we will see next, this translation is not straightforward and requires the construction of novel imaging architectures.

3.3 Optical Krylov Subspaces for Hyperspectral Imaging

In this section, we provide a high-level description of optical computing of HSIs using Krylov subspace methods.

Notation. We represent HSIs in two different ways:

- $H(x, y, \lambda)$ — a real-valued function over 2D space (x, y) and 1D spectrum λ,
- $X \in \mathbb{R}^{N_x N_y \times N_\lambda}$ — a matrix with $N_x N_y$ rows and N_λ columns, such that each column corresponds to the vectorized image at a specific spectrum.

The goal is to optically build the following two operators:

- *Spectrally-coded imager* $\mathcal{I}$ — Given a spectral code $\mathbf{x} \in \mathbb{R}^{N_\lambda}$, we seek to measure the image $\mathbf{y} \in \mathbb{R}^{N_x N_y}$ given as

$$\mathbf{y} = \mathcal{I}(\mathbf{x}) = X\mathbf{x}. \tag{3.1}$$

  The image $\mathbf{y}$ corresponds to a grayscale image of the scene with a camera whose spectral response is $\mathbf{x}$.

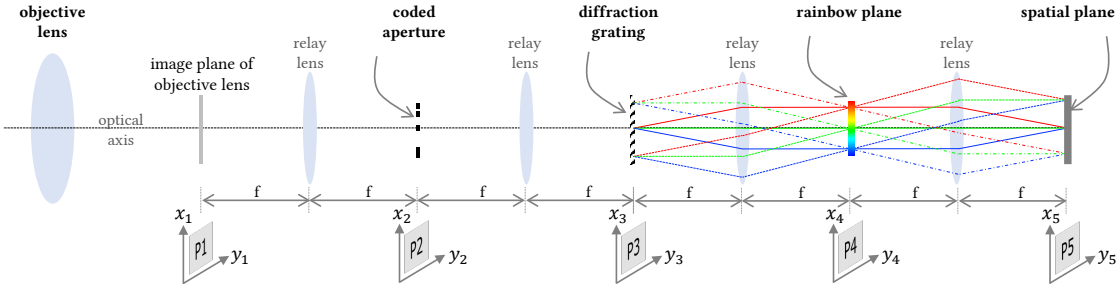


Figure 3.3: **Schematic for simultnenous spatio-spectral measurements with a coded aperture**. The diffraction grating disperses light along x-axis. The image of the scene is formed on plane P1. The coded aperture is placed in P2, which introduces a diffraction blur in spatial plane P3, and dictates the spectral profile formed on the plane P4. A slit or an open aperture on P2 is not a good choice for simultaneously high spatial and spectral resolution. Instead, we rely on design of a novel pupil aperture that enables simultaneous high spatial and spectral resolution.

- *Spatially-coded spectrometer* $\mathcal{S}$ — Given a spatial code $\widetilde{\mathbf{x}} \in \mathbb{R}^{N_x N_y}$, we seek to measure a spectral measurement $\widetilde{\mathbf{y}} \in \mathbb{R}^{N_\lambda}$ given as

$$\widetilde{\mathbf{y}} = \mathcal{S}(\widetilde{\mathbf{x}}) = X^\top \widetilde{\mathbf{x}}. \tag{3.2}$$

  The measurement $\widetilde{\mathbf{y}}$ corresponds to the spectral measurement of the scene, where-in the spectral profile of each pixel is weighted by the corresponding entry in the spatial code $\widetilde{\mathbf{x}}$.

Since the two operators correspond to left and right multiplication of a vector to the HSI matrix X, we can implement any Krylov subspace technique to estimate the top singular vectors and values.

Number of measurements required. To obtain a rank-k approximation of the matrix X, we would require at least k spatially-coded spectral measurements — each of dimensionality N_λ, and k spectrally-coded images — each of dimensionality $N_x N_y$. Hence, the number of measurements required by the approach is proportional to $k(N_x N_y + N_\lambda)$ and, over traditional Nyquist sampling, it represents a reduction in measurements by a factor of

$$\frac{k(N_x N_y + N_\lambda)}{N_x N_y N_\lambda} = k\left(\frac{1}{N_\lambda} + \frac{1}{N_x N_y}\right). \tag{3.3}$$

For low-rank HSIs, we can envision dramatic reductions in measurements required over Nyquist sampling especially when sensing at high spatial and spectral resolutions (see Table 3.1).

Challenges in implementing operators $\mathcal{I}$ and $\mathcal{S}$. Spatially-coded spectral measurements have been implemented in the context of compressive hyperspectral imaging [93]. Here, light from a scene

is first focused onto an SLM that performs spatial coding, and then directed into a spectrometer. For spectral coding at a high-resolution, we could replace the sensor in a spectrometer with an SLM; subsequently, we can form and measure an image of the coded light using a lens. However, high-resolution spectrometers invariably use a slit aperture that produces a large one-dimensional blur in the spatial image due to diffraction. We show in Section 3.4 that simultaneous spatio-spectral localization is not possible with either a slit or an open aperture. This leads to the design of optimal binary coded apertures which enable high spectral and spatial resolutions. Subsequently, in Section 3.6, we present the design of KRISM and validate its performance in Section 3.7.

3.4 Coded apertures for simultaneous sensing of space and spectrum

In this section, we introduce an optical system capable of simultaneously resolving space and spectrum at high resolutions.

3.4.1 Optical setup

The ideas proposed in this chapter rely on the optical setup shown in Figure 3.3 which is a slight modification of a traditional spectrometer. An objective lens focuses a scene onto its image plane, that we denote as P1. This is followed by two $4f$ relays with a coded aperture placed on the first pupil plane, P2, and a diffraction grating placed at the plane marked as P3. We are interested in the intensity images formed at the planes marked at the "rainbow plane" P4 and the "spatial plane" P5, and their relationship to the image formed on P1, the coded aperture, and the grating parameters.

We assume that the field formed on the plane P1 is *incoherent* and, hence, we only need to consider its intensity and how it propagates, and largely ignore its phase. Let $H(x, y, \lambda)$ be the intensity of the field as a function of spatial coordinates (x, y) and wavelength λ. Let $a(x, y)$ be the aperture code placed at the plane P2, v_0 be the density (measured in grooves per unit length) of the diffraction grating in P3, and f be the focal length of the lenses that form the $4f$ relays. The hyperspectral field intensity at the plane P4 is given as

$$F_4(x, y, \lambda) = \frac{1}{\lambda^2 f^2} a^2(-x + f\lambda v_0, -y)\ S(\lambda), \tag{3.4}$$

where $S(\lambda)$ is the scene's overall spectral content defined as

$$S(\lambda) = \int_x \int_y H(x, y, \lambda) dx dy.$$

The intensity field at the spatial plane P5 is given as

$$F_5(x, y, \lambda) = H(x, y, \lambda) * \left| \frac{1}{\lambda^2 f^2} A\left(-\frac{x}{\lambda f}, -\frac{y}{\lambda f}\right) \right|^2, \tag{3.5}$$

where $A(u, v)$ is the 2D spatial Fourier transform of the aperture code $a(x, y)$, and $*$ denotes two-dimensional spatial convolution along x and y axes. These expressions arise from Fourier optics [35] and their derivation is provided in the supplemental material.

Image formed at the rainbow plane P4. A camera with spectral response $c(\lambda)$ placed at the rainbow plane would measure

$$\begin{aligned} I_R(x, y) &= \int_\lambda a^2(-x + f\lambda v_0, -y) \frac{1}{\lambda^2 f^2} S(\lambda) c(\lambda) d\lambda \\ &\propto a^2(-x, -y) * \left(S\left(\frac{x}{f v_0}\right) \widetilde{c}\left(\frac{x}{f v_0}\right) \right), \end{aligned} \tag{3.6}$$

where $\widetilde{c}(\lambda) = c(\lambda)/\lambda^2 f^2$. Here, the dimensionless term $f v_0$, that scales of the spectrum $S(\cdot)$, indicates the resolving power of the diffraction grating. For example, we used a focal length $f = 100$ mm and a grating with groove density $v_0 = 300$ grooves/mm for the prototype discussed in Section 3.6; here, $f v_0 = 30,000$. This implies that the spectrum is stretched by a factor of $30,000$. Therefore, a 1 nm of the spectrum maps to 30 μm, which is about 6-7 pixel-widths on the camera that we used. The key insight this expression provides is that the image I_R is the convolution of the scene's spectrum — denoted as a 1D image — with the aperture code $a(\cdot, \cdot)$ (see Figure 3.4). This implies that we can measure the spectrum of the scene, albeit convolved with the aperture code on this plane; this motivates our naming of this plane as the rainbow plane.

Image at the spatial plane P5. A camera with the spectral response $c(\lambda)$ placed at the spatial plane P5 would measure

$$I_S(x, y) = \int_\lambda \left(H(x, y, \lambda) * \left| \frac{1}{\lambda^2 f^2} A\left(-\frac{x}{\lambda f}, -\frac{y}{\lambda f}\right) \right|^2 \right) c(\lambda) d\lambda \tag{3.7}$$

I_S is a "spatial image" in that spectral components of the HSI have been integrated out. Hence, we refer to P5 as the spatial plane. Figure 3.4 shows the image formed at P5 for different choices of the coded apertures, including slits and open apertures.

Implementing KRISM operations. The derivation above suggests that we get a spatial image of the scene formed at the spatial plane P5 and a spectral profile at the rainbow plane P4. We can therefore build the two operators central to KRISM by coding light on one of the planes while measuring it at

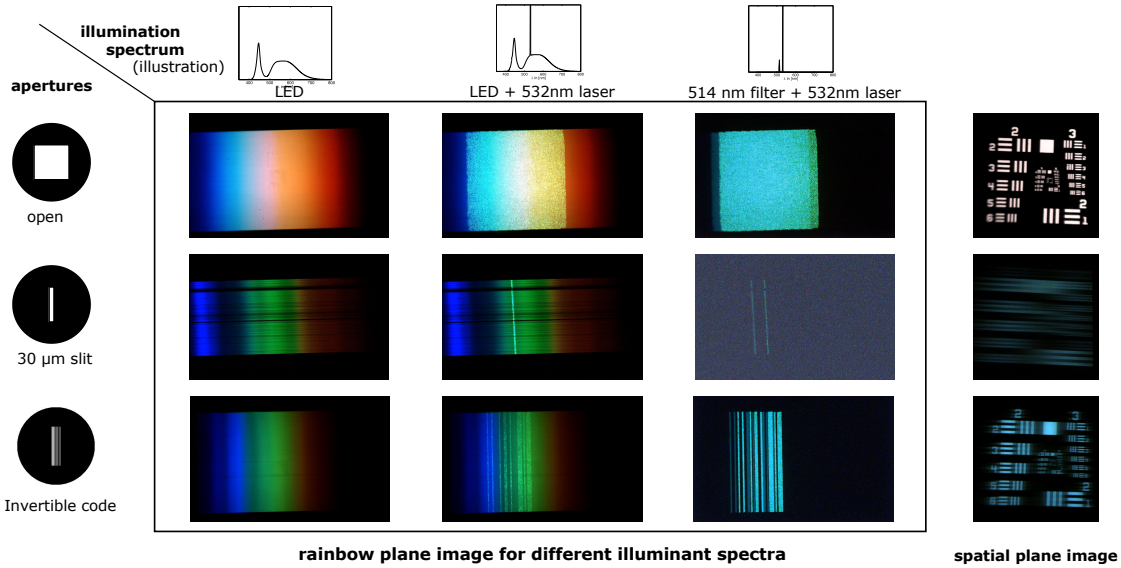
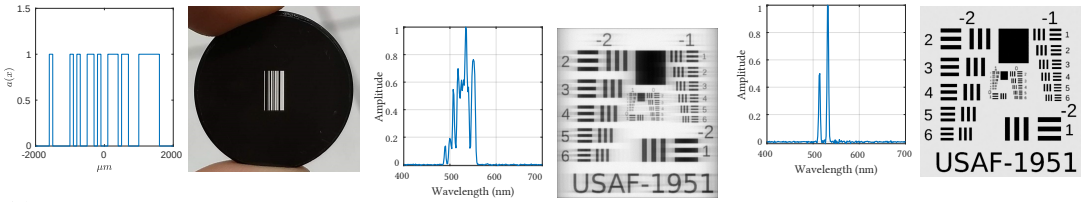


Figure 3.4: **Effect of pupil code on spatial and spectral resolutions**. We implemented the setup shown in Figure 3.3 to verify the effect of different pupil codes. The scene consists of a resolution chart illuminated by two distinct narrowband light sources. An open aperture leads to sharp spatial images, but the spectrum is blurred. On the other hand, a slit offers high spectral resolution, but the spatial image is blurred. Optimized codes offer invertible spectral blur, and at the same time, invertible spatial blur.

the other. For the spectrally-coded imager $\mathcal{I}$, we will place an SLM on the rainbow plane P4 while measuring the image, with a camera, at P5. For the spatially-coded spectrometer $\mathcal{S}$, we place an SLM on P3 — which is optically identical to P5 — while measuring the image formed at P4.

Effect of the aperture code on the scene's HSI Introducing an aperture code $a(x, y)$ on the plane P2 can be interpreted as distorting the scene's HSI in two distinct ways. First, a spectral blur is introduced whose point spread function (PSF) is a scaled copy of the aperture code $a(x, y)$. Second, a spatial blur is introduced for each spectral band whose PSF is the power spectral density (PSD) of the aperture code, suitably scaled. With this interpretation, the images formed on planes P4 and P5 are a spectral and spatial projection, respectively, of this new blurred HSI. Our proposed technique measures a low-rank approximation to this blurred HSI and we can, in principle, deblur it to obtain the true HSI of the scene. However, the spatial and spectral blur kernels may not always be invertible. As we show next, the choice of the aperture is critical and that traditional apertures such as a slit in spectrometry and an open aperture in imaging will not lead to invertible blur kernels.



(a) Optimized code and aperture for $N = 32$.

(b) Raw measurements.

(c) Deconvolved spectrum and image respectively.

Figure 3.5: **Optimized codes for high spatial and spectral resolution**. Optimized codes ensure that the spectral as well as spatial blur can be deconvolved stably. We simulate the performance of optimal code on a spatial and spectral target similar to Figure 3.4. Spectrum was deconvolved using Wiener deconvolution, and spatial images were deconvolved using TV prior. Optimized codes offer high spatial as well as spectral resolution.

3.4.2 Failure of slits and open apertures

We now consider the effect of the traditional apertures used in imaging and spectrometry — namely, an open aperture and a slit, respectively — on the images formed at the rainbow and the spatial planes. Suppose that the aperture code $a(x, y)$ is a box function of width W mm and height H mm, i.e.,

$$a(x, y) = \text{rect}_W(x)\,\text{rect}_H(y).$$

Its Fourier transform $A(u, v)$ is the product of two sincs

$$A(u, v) = \text{sinc}(Wu)\,\text{sinc}(Hv).$$

The spatial image I_S is convolved with the PSD $|A(u, v)|^2$ scaled by $f\lambda$, so the blur observed on it has a spatial extent of $f\lambda/W \times f\lambda/H$ units. Suppose that $f = 100$ mm and $\lambda = 0.5\mu$m, the observed blur is $50/W \times 50/H$ $(\mu\text{m})^2$. The rainbow plane image I_R, on the other hand, simply observes a box blur whose spatial extent is $W \times H$ mm^2. Armed with these expressions, we can study the effect of an open and a slit apertures on the spatial and rainbow images.

Scenario #1 — An open aperture. Suppose that $W = H = 10$ mm, then we can calculate the spatial blur to be 5μm in both height and width and hence, we can expect a very sharp spatial image of the scene. The blur on the rainbow image has a spread of 10 mm; for relay lenses with focal length $f = 100$mm and grating with groove density $v_0 = 300$ grooves/mm, this would be equivalent of a spectral blur of $10,000/30 \approx 333$ nm. Hence, we cannot hope to achieve high spectral resolution with an open aperture.

Scenario #2 — A slit. A slit is commonly used in spectrometers; suppose that we use a slit of width $W = 100\mu$m and height $H = 10$mm. Then, we expect to see a spectral blur of $100/30 \approx 3.3$ nm. The spatial image is blurred along the y-axis by a 5μm blur and along the x-axis by a $50/0.1 = 500\mu$m blur; effectively, with a 5μm pixel pitch, this would correspond to a 1D blur of 100 pixels. In essence, the use of a slit leads to severe loss in spatial resolution.

Figure 3.4 shows images formed at the rainbow and spatial planes for various aperture codes. This validates our claim that conventional imagers are unable to simultaneously achieve high spatial and spectral resolutions due to the nature of the apertures used. We next design apertures with carefully engineered spectral and spatial blurs, which can be deblurred in post-processing.

3.4.3 Design of aperture codes

We now design an aperture code that is capable of resolving both space and spectrum at high-resolutions. Our use of coded apertures is inspired by seminal works in coded photography for motion and defocus deblurring [53, 82, 97].

Observation. Recall that the rainbow plane image I_R is a convolution between a 1D spectral profile $s(\cdot)$ and a 2D aperture code $a(x, y)$. This convolution is one dimensional, i.e., along the x-axis; hence, we can significantly simplify the code design problem by choosing an aperture of the form

$$a(x, y) = a(x)\ \mathrm{rect}_H(y), \tag{3.8}$$

with H being as large as possible. The choice of the rect function along the y-axis leads to a high light throughput as well as a compact spatial blur along the y-axis. For ease of fabrication, we further restrict the aperture code to be binary and of the form

$$a(x) = \sum_{k=0}^{N-1} a_k\, \mathbb{I}_{[k\Delta,(k+1)\Delta]}(x), \tag{3.9}$$

where $\mathbb{I}_{[p,q]}(x) = 1$ when $x \in [p, q]$ and zero otherwise. Hence, the mask design reduces to finding an N-bit codeword $\mathbf{a} = \{a_0, \ldots, a_{N-1}\}$. The term Δ, with units in length, specifies the physical dimension of each bit in the code. We fix its value based on the desired spectral resolution. For example, for $f = 100$mm and $v_0 = 300$ grooves/mm, a desired spectral resolution of 1nm would require $\Delta \leq 30\mu$m.

Our goal is to design masks that enable the following:

- *High light throughput.* For a given code length N, we seek codes with large light throughput which is equal to the number of ones in the code word $\mathbf{a}$.

- *Invertibility of the spatial and spectral blur.* The code is designed such that the resulting spatial and spectral blur are both invertible.

An invertible blur can be achieved by engineering its PSD to be flat. Given that the spectrum is linearly convolved with $a(x)$, a $(N + N_\lambda - 1)$-point DFT of the code word $\mathbf{a}$ captures all the relevant components of the PSD of $a(x)$. Denoting this $(N + N_\lambda - 1)$-point DFT of $\mathbf{a}$ as $\mathbf{A}[k]$, we aim to maximize its minimum value in magnitude. Recall from (3.7) that the spatial PSF is the power spectral density (PSD) of $a(x)$, with suitable scaling. Specifically, the Fourier transform of spatial blur is given by $c(\lambda f u)$, where $c(x) = a(x) * a(-x)$ is the linear autocorrelation of $a(x)$ and u represents spatial frequencies. From (3.9), we get,

$$\begin{aligned} c(x) &= a(x) * a(-x) \\ &= \sum_{k=-N}^{N-1} c_k \left(\mathbb{I}_{[k\Delta,(k+1)\Delta]}(x) * \mathbb{I}_{[k\Delta,(k+1)\Delta]}(x) \right), \end{aligned} \tag{3.10}$$

where c_k is the discrete linear autocorrelation of a_k. Thus, it is sufficient to maximize c_k to obtain an invertible spatial blur.

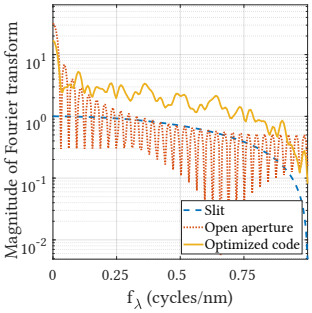
We select an aperture code that leads to invertible blurs for both space and spectrum by solving the following optimization problem:

$$\max_{a_0,\ldots,a_{N-1}} \alpha \min_k \left(|\mathbf{A}[k]| \right) + (1 - \alpha) \min_k c_k, \tag{3.11}$$

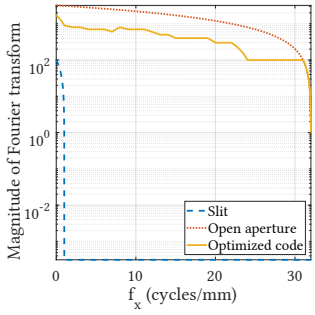
under the constraint that the elements of $\mathbf{a}$ are binary-valued, and $\alpha \in (0, 1)$ is a constant. For code length N sufficiently small, we can simply solve for the optimal code via exhaustive search of all $2^N - 1$ code words. We used $N = 32$ and an exhaustive search for the optimal code took over a day. The resulting code and its performance is shown in Figure 3.5 and 3.6; we used $\Delta = 100\mu$m and $H = 6.4$mm for this result. A brute force optimization is not scalable for larger codes. Instead of searching for optimal codes, we can search for approximately optimal codes by iterating over a few candidate solutions. This strategy has previously been explored in [82], where 6 million candidate solutions are searched for a 52-dimensional code.

Figure 3.6 shows the frequency response of both spectral and spatial blurs for the 32-dimensional optimized code. The advantages of optimized codes are immediately evident — an open aperture has several nulls in spectral domain, while a slit attenuates all high spatial frequencies. The optimized code retains all frequencies in both domains, while increasing light throughput.

We now discuss alternate designs for the coded aperture and the specific algorithm we used for deconvolving the spatial and spectral measurements.



(a) Freq. response of spectral blur



(b) Freq. response of spatial blur

Figure 3.6: **Frequency response of spatial and spectral blur for various pupil codes**. Width of the slit was 100μm, while that of open aperture was 3.2mm. The length of optimized code is 32-bits, with each bit being 100μm wide, giving a 3.2mm wide aperture. We assume that a slit can resolve up to 1nm. In the graph, 0.5 cycles/nm corresponds to a spectral resolution of 1nm, and hence the frequency response of the slit falls off after 0.5 cycles/nm. Similarly, the maximum spatial resolution is $15\mu m$ and hence f_x is shown till 32 cycles/mm. For spectral measurements, a slit has a flat frequency response, while an open aperture has several nulls. In contrast, an open aperture has no nulls for spatial measurements, whereas a slit attenuates high frequencies. Optimized codes have a fairly flat frequency response for spectral blur, and no nulls for spatial blur.

3.4.4 Other code designs

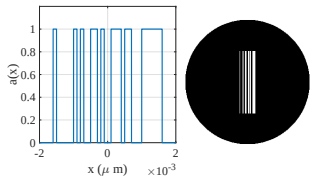
We now show other choices for coded apertures and compare their performance.

Spatially compact codes

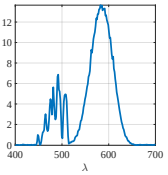
Instead of pursuing spatially invertible codes, we can optimize for codes which introduce compact spatial blur. In such a case, the goal would be to suppress side lobes of the PSF of spatial PSF. Let $P_a(x)$ be the spatial PSF created by $a(x)$. If η_1, η_2 be the first and second maximum peak heights of $P_a(x)$, then maximizing the ratio η_2/η_1 leads to spatially imperceptible blur. Combined with an invertible spectral blur, we formulate the overall objective function as:

$$\max_{a_1,\ldots,a_N} \alpha \min_k (|\mathbf{A}[k]|) + (1-\alpha)\left(\frac{\eta_2}{\eta_1}\right), \tag{3.12}$$

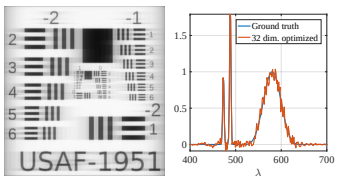
where $\alpha \in (0,1)$ is a constant. As with optimized codes, we brute forced the optimal solution.



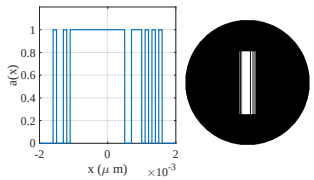
(a) Optimized code and aperture for $N = 32$.



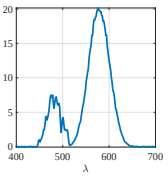
(b) Raw measurements.



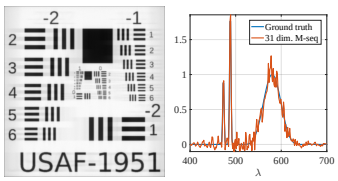
(c) Deconvolved spectrum and image respectively (Spectrum PSNR: 31.9dB; Spatial PSNR: 27.2dB)



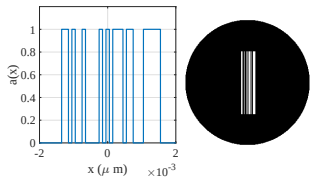
(d) Spatially-compact code and aperture for $N = 32$.



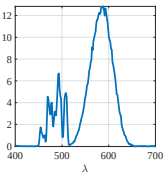
(e) Raw measurements.



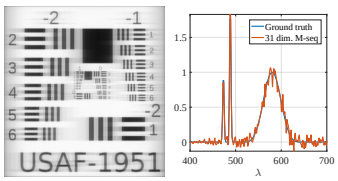
(f) Deconvolved spectrum and image respectively (Spectrum PSNR: 29.0dB; Spatial PSNR: 26.4dB)



(g) M-sequence code and aperture for $N = 31$.



(h) Raw measurements.

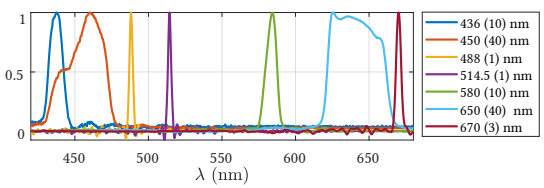


(i) Deconvolved spectrum and image respectively (Spectrum PSNR: 29.4dB; Spatial PSNR: 26.4dB)

Figure 3.7: **Comparison of performance of various codes**. We compare optimized codes, spatially compact codes and M-sequences. Simulations were performed with added readout and poisson noise. Spectral deconvolution was done with Wiener deconvolution, while spatial deconvolution was done with TV-prior. While spatially-compact codes (row 2) offer better performance in spatial images, spectral deconvolution accuracy is very low. M-sequences (row 3) perform moderately well for both spectral and spatial deconvolution. However, optimized codes perform the best overall (row 1).

M-sequences

Maximal length sequences, or M-sequences for short, are optimal codes when using circular convolution. Their PSD is flat and hence is desirable as blur functions. However, since our convolution is linear, M-

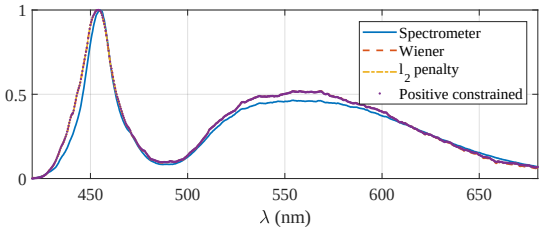


(a) Spectra of narrowband filters.

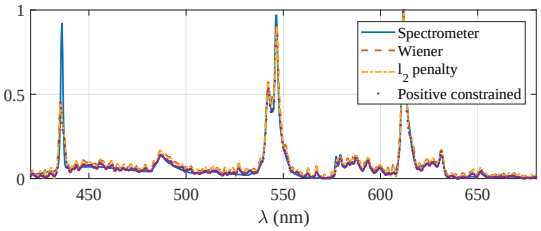
True central wavelength (nm)	Measured central wavelength (nm)	True FWHM (nm)	Measured FWHM (nm)
436	437.0	10	10.9
450	456.1	40	31.8
488	488.3	1	2.6
514.5	514.3	1	2.3
580	583.6	10	8.2
650	641.6	40	40.4
670	670.0	3	3.7

(b) Central wavelength and FWHM of the filters

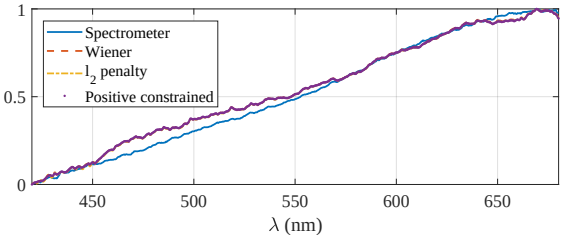
Figure 3.8: **Calibration results with narrowband spectral filters**. Spectra of some narrowband filters with datasheet central wavelength and FWHM (in parenthesis) provided in legend. We used Weiner deconvolution to obtain the true spectra. (b) tabulates the estimated central wavelength and FWHM, along with datasheet values. The accuracy of central wavelength and FWHM establishes the accuracy as well as high-resolution capabilities of our optical setup.



(a) White Light Emitting Diode light source



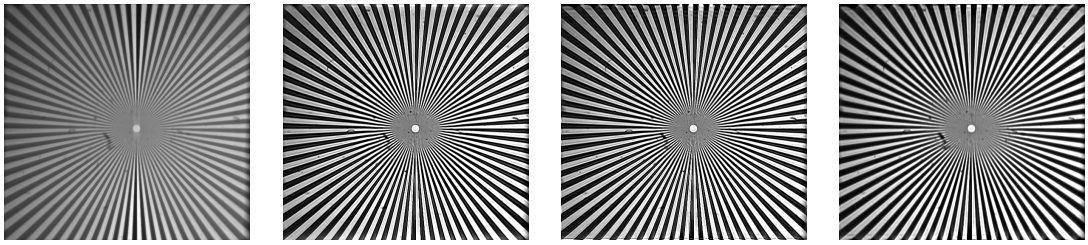
(b) White Compact Fluorescent Lamp



(c) Tungsten-halogen light source.

Figure 3.9: **Spectra of commonly found light sources**: (a) Tungsten-halogen, (b) Light Emitting Diode (LED), and (c) Compact Fluorescent Lamp (CFL). Results are shown for Wiener deconvolution, ℓ_2 penalized deconvolution and positive-constrained deconvolution. CFL shows some error in blue wavelengths, as the machine vision camera we used was not reliable for deep blue wavelengths. The deconvolved results are robust to choice of algorithm, as the aperture code was designed to be invertible.

sequences are not necessarily the optimal choice.



(a) Before deconvolution **(b)** TV-prior deconvolution **(c)** Wiener deconvolution **(d)** Richardson-Lucy deconvolution

Figure 3.10: **Deconvolution results of a Siemen star with various deconvolution algorithms**. Due to invertible nature of PSF, deconvolution is robust to choice of method. However, TV-prior gave best results in terms of higher contrast ratio.

3.4.5 Performance comparison

We compare optimized codes, spatially compact codes and M-sequences for their performance in spatial deconvolution and spectral deconvolution. To test spectral deconvolution, we created a spectrum with two closely spaced narrowband peaks and a broadband peak, and blurred them with various codes. Readout noise and shot noise were added to adhere to real world measurements. Finally, deconvolution was done with wiener filter. To test spatial deconvolution, we used Airforce target and blurred with the scaled PSD of the pupil codes, and added noise. Deconvolution was done with a TV prior in all cases.

Figure 3.7 shows a comparison of performance for spectral and spatial deconvolution. As expected, optimized codes perform the best for spectral deconvolution, while spatially compact codes perform worse. Spatially compact codes perform the best in this case, while optimized codes come close. Since hyperspectral imaging requires good spatial as well as spectral resolution, we chose optimized codes.

3.4.6 Spectral deconvolution

Recall that our optical setup measures a blurred version of the true spectrum. Specifically, if the aperture code is $a(x)$ and the spectrum to be measured is $s(\lambda)$, our optical setup measures $y(\lambda) = a(\lambda) * s(\lambda) + n(\lambda)$, where $n(\lambda)$ is additive white gaussian noise. The addition of noise prevents us from simply dividing in Fourier domain. Fortunately, since the aperture code was designed to be invertible, it is fairly robust to noise. A naive solution, such as Wiener deconvolution, hence, works very well. If the noise is too high, or the spectra is known to be smooth, we can impose an ℓ_2 penalty on the difference and solve the

following optimization problem:

$$\min_{\mathbf{s}} \quad \frac{1}{2}\|\mathbf{y} - \boldsymbol{\alpha} * \mathbf{s}\|^2 + \eta\|\nabla\mathbf{s}\|^2, \tag{3.13}$$

where $\mathbf{y}$ is the measured spectrum, $\boldsymbol{\alpha}$ is the aperture code, $\mathbf{s}$ is the spectrum to be recovered, and $\nabla\mathbf{s}$ is the first difference of $\mathbf{s}$. Further priors, such as positivity constrains give better results as well. Figure 3.9 shows a comparison of spectra of various commonly available light sources, as well as a comparison with spectrometric measurements. We showed results for three forms of deconvolution, namely, Wiener deconvolution, ℓ_2 regularized deconvolution, and positivity constrained deconvolution. Figure 3.8 shows results for some narrowband filters. We computed the central wavelength and Full Width Half Max (FWHM) for each filter and compared it against the numbers provided by the company. As expected, the FWHM of 1nm filters is between 2nm and 3nm, as the FWHM of our optical setup is 3nm. FWHM for 10nm filters and 40nm filters is close to the ground truth values.

3.4.7 Spatial deconvolution

The presence of a coded aperture introduces a blur in spatial domain, which is the scaled power spectral density of the coded aperture. Our optimization procedure accounts for invertible spectral blur as well as invertible spatial blur. Hence, deconvolution is stable even in the presence of noise. While naive deconvolution procedures such as Wiener deconvolution or Richardson-Lucy work well, we imposed total variance (TV) penalty on the edges to get more accurate results. Figure 3.10 shows blurred image of Siemen star, and deconvolution with TV-prior, Wiener and Richardson-Lucy algorithms. As with spectral deconvolution, spatial deconvolution is fairly robust to choice of algorithm. We chose TV-prior, as it returned the sharpest results. Figure 3.11 shows MTF before and after deconvolution, with TV-prior. Deconvolution signficantly improves the MTF30 value, which jumps from 20 line pairs/mm to 90 line pairs/mm.

3.5 Synthetic experiments

We tested KRISM via simulations on three different datasets, listed in Table 3.2, and compared against existing approaches. For all methods, we simulated both photon and readout noise respectively as Poisson and Gaussian random variables. All KRISM simulations were done with diffraction effects due to coded aperture.

We quantify performance through compression in measurements N/M which is ratio of number of unknowns to measurements and peak signal to noise ratio (PSNR). Given a HSI matrix $\mathbf{x}$ and its

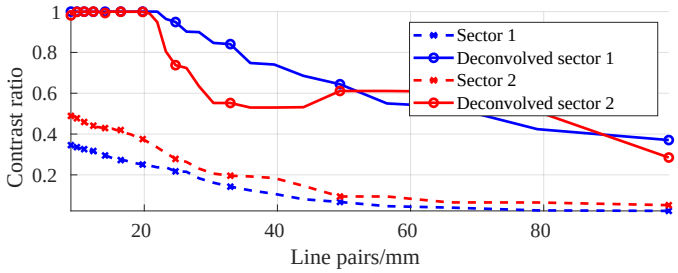


Figure 3.11: **MTF plot before and after deconvolution**. PSF was estimated by capturing image of a $10\mu m$ pinhole. Deconvolution was then done using a TV prior on the image gradients. There is a marked improvement in contrast ratio after deconvolution. The MTF30 value for both sectors jumps from 20 line pairs/mm to 90 line pairs/mm.

reconstruction $\widehat{\mathbf{x}}$, we define peak SNR as

$$\text{PNSR} = 20\log_{10}\left(\frac{\|\mathbf{x}\|_\infty}{\text{RMSE}(\mathbf{x},\widehat{\mathbf{x}})}\right),$$

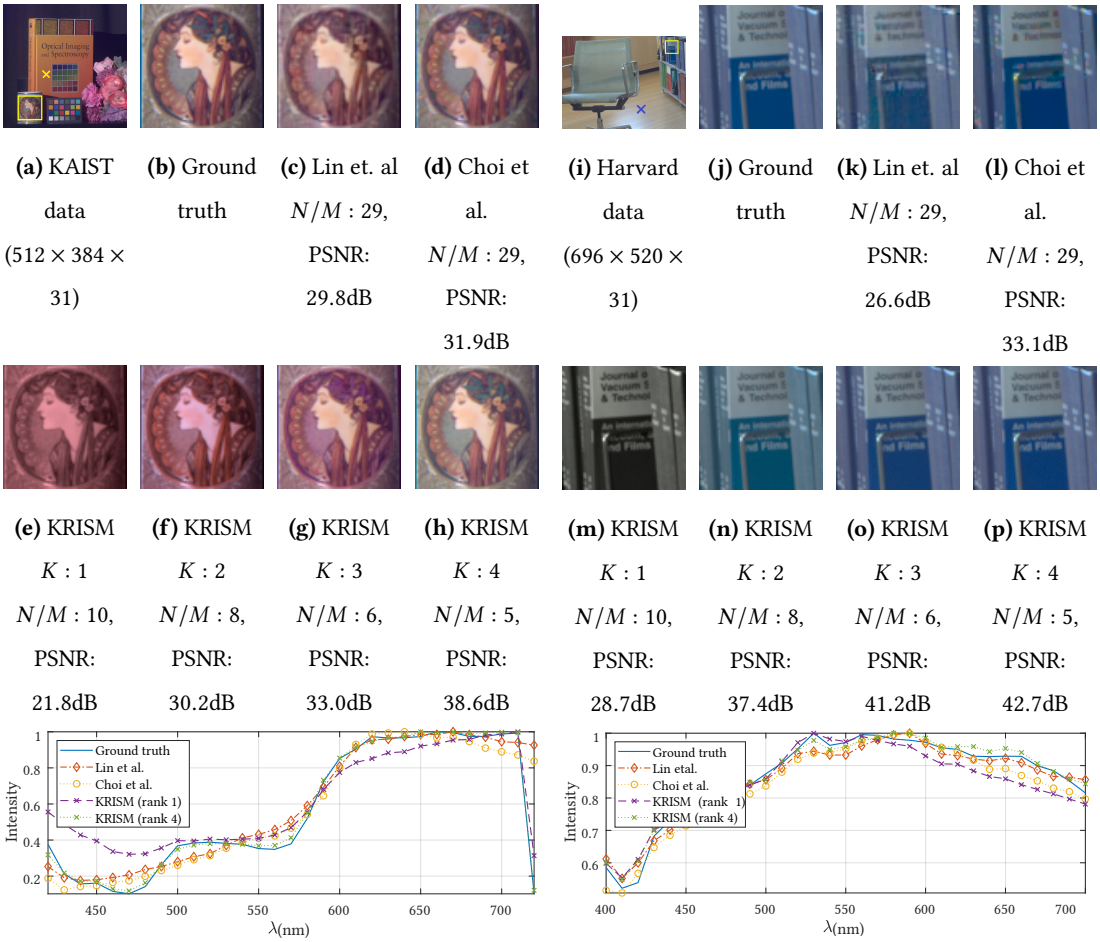
where RMSE is the root mean squared error defined as

$$\text{RMSE}(\mathbf{x},\widehat{\mathbf{x}}) = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \widehat{\mathbf{x}}_n)^2}. \tag{3.14}$$

Dataset	Spatial resolution	#Spectral bands	Waveband (nm)
KAIST [Choi et al. 2017]	512 x 384	31	400-700
Harvard [Charkabarti et al. 2011]	696 x 520	31	400-700
ICVL [Arad and Ben-Shahar 2016]	256 x 256	260	390-1043

Table 3.2: **Datasets used for simulations**. The spatial resolution for KAIST and Hardvard datasets, and spectral resolution for ICVL dataset was reduced to keep computation tractable with competing methods.

Comparison with snapshot techniques. Snapshot techniques such as CASSI [98] and spatial-spectral encoded CS [57] recover HSI from a single image and hence are appropriate for video-rate hyperspectral imaging. In contrast, KRISM is *not* a snapshot technique since, at the very least it requires the measurement of an image and a spectral profile. Nevertheless, we compare KRISM against snapshot techniques by varying the number of KRISM iterations. Figure 3.12 shows performance of these methods with varying number of measurements on KAIST and Harvard datasets. We observe that in the setting closest to snapshot mode, Choi *et al.* [18] and Lin *et al.* [57] do outperform KRISM; this is to be expected



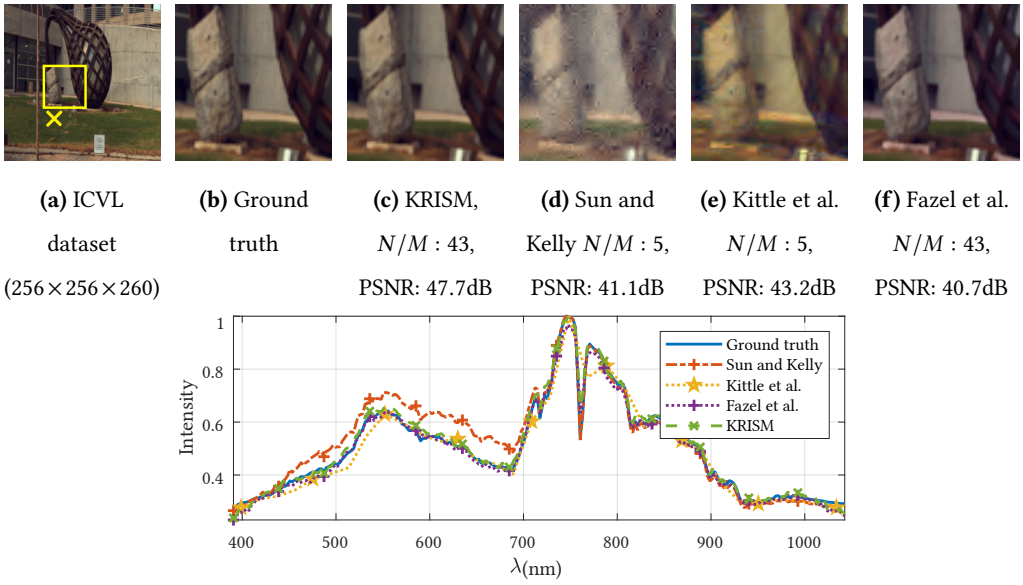
(a) KAIST data ($512 \times 384 \times 31$) **(b)** Ground truth **(c)** Lin et. al N/M : 29, PSNR: 29.8dB **(d)** Choi et al. N/M : 29, PSNR: 31.9dB **(i)** Harvard data ($696 \times 520 \times 31$) **(j)** Ground truth **(k)** Lin et. al N/M : 29, PSNR: 26.6dB **(l)** Choi et al. N/M : 29, PSNR: 33.1dB

(e) KRISM K : 1 N/M : 10, PSNR: 21.8dB **(f)** KRISM K : 2 N/M : 8, PSNR: 30.2dB **(g)** KRISM K : 3 N/M : 6, PSNR: 33.0dB **(h)** KRISM K : 4 N/M : 5, PSNR: 38.6dB **(m)** KRISM K : 1 N/M : 10, PSNR: 28.7dB **(n)** KRISM K : 2 N/M : 8, PSNR: 37.4dB **(o)** KRISM K : 3 N/M : 6, PSNR: 41.2dB **(p)** KRISM K : 4 N/M : 5, PSNR: 42.7dB

Figure 3.12: **Evaluation against snapshot techniques**. We compare KRISM with varying rank against results from [57] and [18] in terms of compression as well as accuracy. We show zoomed in image patches for each method and spectrum at pixel marked by a cross. At similar compression rates ($K = 1$), KRISM has lower accuracy than snapshot techniques. However, snapshot techniques require solving a complex optimization problem that can be time consuming. In contrast, KRISM requires practically no reconstruction time as the dominant singular vectors are captured directly.

since after a single iteration, KRISM provides only a rank-1 approximation. As the number of KRISM iterations are increased (which allows approximations of higher ranks), KRISM performance improves. KRISM enjoys advantages when we look at computational cost for reconstruction. The reconstruction time for Choi *et al.* [18] is more than 10 minutes[1] even with multiple GPUs, while it runs to several

[1]We used code, dataset and model from https://github.com/KAIST-VCLAB/deepcassi



(a) ICVL dataset ($256 \times 256 \times 260$)

(b) Ground truth

(c) KRISM, N/M : 43, PSNR: 47.7dB

(d) Sun and Kelly N/M : 5, PSNR: 41.1dB

(e) Kittle et al. N/M : 5, PSNR: 43.2dB

(f) Fazel et al. N/M : 43, PSNR: 40.7dB

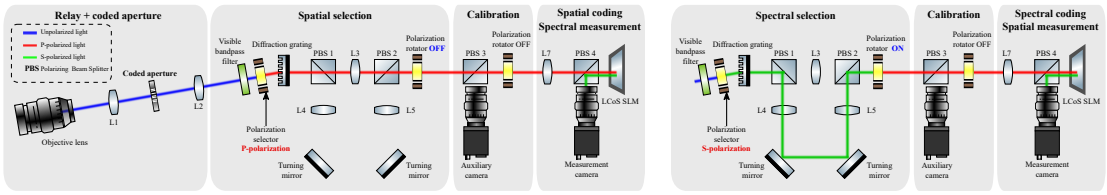
Figure 3.13: **Evaluation against multi-frame techniques**. We compare KRISM against spatially-multiplexed HSI [93], multi-frame version of CASSI [49], and row/column CS [27]. We show zoomed in image patches for each method and spectrum at pixel marked by a cross. Across the board, KRISM has highest accuracy with fewest measurements.

hours for Lin *et al.* [57][2]. In contrast, KRISM requires practically no reconstruction time for recovering the HSI as we directly measure the singular vectors.

Comparison with multi-frame techniques. Since KRISM is essentially a multi-frame technique, we compare against multi-frame version of CASSI [49], and spatially-multiplexed hyperspectral imager [93]. We simulate spatially-multiplexed HSI imager via randomly permuted Hadamard multiplexed spectra and recover using sparsity of individual bands in wavelet domain. Note that the compression ratio is lower for Kittle *et al.* [49] and Sun and Kelly [93] since the results were inaccurate for higher compressions[3]. Figure 3.13 shows a comparison of recovered spatial and spectral images for ICVL dataset. The poor performance of Kittle *et al.* [49] is due to usage of a translational mask to get multiple measurements. On the other hand, Sun and Kelly [93] performs poorly as multiplexing is done only in the spatial domain. Performance can be improved if we multiplex in the spectral domain as well; the resulting method is the low-rank CS approach proposed by Fazel *et al.* [27]. This results in an increase in accuracy with fewer measurements, as seen in Figure 3.13 (f). Note that CS-based techniques are

[2]We used code, dataset and overcomplete dictionary from the paper itself.

[3]Please see supplementary for further details.



(a) Optical setup in spatially-coded spectral measurement mode

(b) Optical setup in spectrally-coded spatial measurement mode

Figure 3.14: **Schematic of KRISM optical setup**. Proposed optical setup in spectral coding (a) and spatial coding (b) mode. The optical method relies on polarization to switch between the two types of coding. When the input light is S-polarized, the LC rotator is switched off, enabling spectrally coded spatial measurements. When the input light instead is P-polarized, the LC rotator is turned on, which enables spatially coded spectral measurements. The input light polarization is controlled by a second LC rotator placed before the grating. With a novel use of LC rotators, our optical setup enables dual coding of hyperspectral scenes with a single camera-SLM pair.

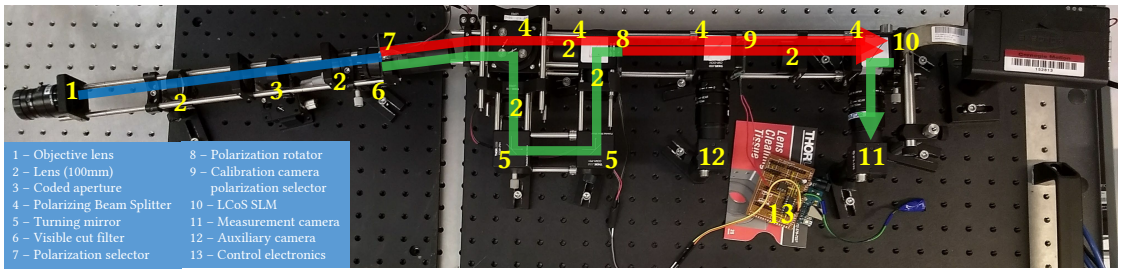


Figure 3.15: **Photograph of our lab prototype**. The optical paths for spectral as well as spatial coding shown in Fig. 3.14 have been overlaid for easy understanding. Components have been marked, grouped and labeled for convenience. All other relevant information is available in supplementary material.

based on random projections and are not adapted to the scene. In contrast, KRISM adaptively computes a low-rank approximation leading to an increase in accuracy with the same number of measurements as Fazel *et al.* [27].

Based on these simulations, we conclude that KRISM is indeed a compelling methodology when spatial and spectral resolution are high — a desirable operating point in many applications. When the number of spectral bands are smaller, the gains are modest, but nevertheless present. In the next section, we provide an optical schematic for implementing KRISM.

3.6 The KRISM Optical setup

We now present an optical design for implementing the two operators presented in Section 3.3 and analyzed in Section 3.4. For efficiencies in implementation, we propose a novel design that combines

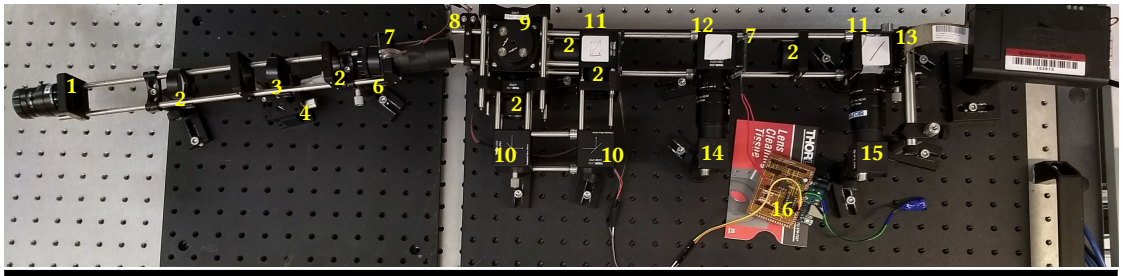
both operators into one compact setup. Figure 3.14 shows a schematic that uses polarization to achieve both operators with a single SLM and a single camera. First, in Figure 3.14(a), an SLM is placed $2f$ away from the grating, and an image sensor $2f$ away from the SLM, implementing spectrally coded spatial measurement operator $\mathcal{I}$. In Figure 3.14(b), light follows an alternate path where in the SLM is $4f$ away from the grating; the camera is still $2f$ away from the SLM. This light path allows us to achieve the spatial-coded spectral measurement operator $\mathcal{S}$. The two light pathways are combined using a combination of polarizing beam splitters (PBS) and liquid crystal rotators (LC). The input light is pre-polarized to be either S-polarized or P-polarized. When the light is P-polarized, the SLM is effectively $2f$ units away from the grating leading to implementation of $\mathcal{I}$, the spectrally-coded imager. When the light is S-polarized, the SLM is $4f$ units away, provided the polarizing beamsplitter, PBS 3 was absent. To counter this, an LC rotator is placed before PBS 3 that rotates S-polarization to P-polarization when switched on. Hence, when S-light is input in conjugation with the rotator being switched on, we achieve the operator $\mathcal{S}$, a spatially-coded spectrometer. By simultaneously controlling the polarization of input light and the LC rotator, we can implement both $\mathcal{I}$ and $\mathcal{S}$ operators with a single camera and SLM pair.

Figure 3.15 shows our lab prototype with the entire light pathway including the coded aperture placed in the relay system between the objective lens and diffraction grating. The input polarization is controlled by using a second LC rotator with a polarizer, placed before the diffraction grating. Finally, an auxiliary camera is used to image the pattern displayed on the SLM. This camera is used purely for alignment of the pattern displayed on the SLM.

Figure 3.16 shows an annotated image of the optical setup we built along with a list of components along with their company and item number. The system was optimized for a central wavelength of 580nm and hence the relay arm till the diffraction grating has been tilted at 10° with respect to the diffraction grating to correct for schiempflug. Lenses in the relay arm are tilted by 5° with respect to the diffraction grating so that the objective can be aligned with the relay arm without any further tilt. The first beamsplitter (component 8) and the second turning mirror (component 10) have been placed on a kinematic platform to correct for misalignments in the cage system. It is of importance that we chose an LCoS instead of a DMD for spatial light modulation. The reasons:

- Since the output after modulation by DMD is not rectilinear to the DMD plane, it introduces further scheimpflug, which is hard to correct.
- DMD acts as a diffraction grating with Littrow configuration, as it is formed of extremely small mirror facets. This will introduce artifacts in measurements which are non-linear.

Some more design considerations are enumerated below:



Index	Item name	Item Number	Index	Item name	Item Number
1	75mm C-mount objective lens	Thorlabs, MVL12M23	9	polarizing beam splitter on kinematic stage	Edmund Optics, 48-545; Thorlabs, C4W cage cube; Thorlabs, B4C kinematic mount
2	100mm achromatic lenses	Thorlabs, AC254-100-A	10	45 degree mirrors	Thorlabs, CCM1-P01
3	coded aperture	Front Range Photomask, LLC, photomask printed with 10,000 DPI on polyster sheet	11	plate polarizing beam splitter in cage	Edmund Optics, 48-545; Thorlabs, CM1-DCH; Thorlabs, FFM1
4	Z-translation stage	Edmund Optics, 66-495	12	polarizing beam splitter cube in cage	Thorlabs, CCM1-4ER; Thorlabs, WPBS254-VIS
5	XY-translation stage	Thorlabs, DT12XY	13	LCoS microdisplay	Holoeye, HES60016
6	visible filter (420-680nm)	Edmund Optics, 89-794	14	auxiliary camera	FLIR, Flea3 13E4M
7	Liquid Crysltal (LC) rotator	Liquid Crystal Technologies, 25mmx36mm LC shutter	15	measurement camera	FLIR, Blackfly BFLY-U3-23S6M-C
8	diffraction grating	Thorlabs, GT13-03	16	electronics for LC rotator control	Arduino Uno; L293D

Figure 3.16: **List of components for the KRISM optical setup**. All the important components, their company and item number have been listed for reference. Construction component names such as cage plates, rods, posts and breadboards have been omitted for brevity.

1. *Lenses.* We used 100mm achromats for all lenses except the last lens before cameras. Achromats were the most compact and economical choice for our optical setup, while offering low spatial and spectral distortion.

2. *Polarizing beam splitters.* We used wire grid polarizing beamsplitters everywhere to ensure low dependence of spectral distortion on angle of incidence, and increase the contrast ratio.

3. *Using an objective lens for measurement camera.* Note that a lens is placed between the LCoS and measurement sensor which converts spatially-coded image to spectrum and coded spectrum to spatial image. Instead of using another achromat, we used an objective lens set to focus at infinity. Since objective lenses are free of any distortions, and are optimized to focus at infinity, this significantly improves resolution of measurements.

4. *Diffraction grating.* We used an off-the-shelf transmissive diffraction grating with 300 groves/mm, which offered most compact spectral dispersion without any overlap with higher orders. This ensured that there would be no spectral vignetting at any point in the setup.

5. *Polarization rotators.* We bought off-the-shelf Liquid Crystal (LC) shutters and peeled off the polarizers on either sides to construct polarization rotators. This is the most economic option, while offering contrast ratios as high as 400:1. The key drawback is that the settling time is 330ms, which prevents their usage at very high rate. A natural workaround is to incorporate binary Ferroelectric shutters which have a low latency rate of 1ms. However, since ours was only a lab prototype, we decided to go with the cheaper option.

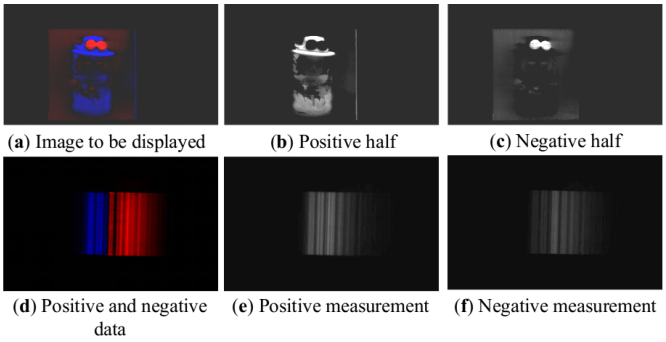
3.6.1 Choice of code size

The pupil code has two free parameters, the length of the code N and the pitch size Δ. The two parameters control the invertibility of spectrum and imperceptibility of spatial images. To understand our design choices, we present constrains and physical dimensions of various measurements. Let each lens in the optical setup have a focal length f and aperture diameter a_L. Let pixel pitch of measurement camera be p. This implies that the camera can capture all spatial frequencies up to $f_{max} = \frac{1}{2p} m^{-1}$. Let the size of grating be a_g in each dimension and its grove density be $g\, groves/mm$.

We capture wavelengths from $\lambda_1 = 420nm$ to $\lambda_2 = 680nm$. The grating equation is given by, $a\sin(\theta) = m\lambda$, where a is the groves spacing and m is the order of diffraction, 1 in our case. Solving for angular spread of spectrum, we get, $\Delta\theta = \sin^{-1}\left(\frac{\lambda_2}{a}\right) - \sin^{-1}\left(\frac{\lambda_1}{a}\right)$ The size of spectrum then is $f\tan(\Delta\theta) + N\Delta$. The minimum resolvable wavelength is $\Delta\lambda \approx \frac{\Delta}{af}$. To avoid vignetting in a $4F$ system, we require that the pupil plane be no larger than $a_L - a_g mm$, giving us $f\tan(\Delta\theta) + N\Delta \leq a_L - a_g$.

Recall that the pupil code is $a(x) = b(x) * \sum_{k=0}^{k=N-1} a[k]\delta(x - k\Delta)$, where $a[k]$ is the binary pupil code and $b(x) = 1 \quad -\Delta/2 \leq x \leq \Delta/2$. Using the formula for PSF of an incoherent system, we know that the Fourier transform of the PSF is $F_{PSF} = C_a(\lambda f u)$, where $C_a(x)$ is the linear autocorrelation of $a(x)$ and u is spatial frequency in $1/m$. To capture all spatial frequencies, we need $F_{PSF}(u)$ to be non-zero for $u \geq f_{max}$, which gives us $N\Delta \geq \lambda f \frac{1}{2p}$.

In our optical setup, we have $a_L = 25mm, a_g = 12.5mm, p = 5\mu m$, $f = 100mm$, $\lambda = 500nm$, and $\Delta = 100\mu m$, which leaves us N, and g as free variables. To prevent vignetting, we need $N\Delta$ to be less than $a_L - a_g - f\tan(\Delta\theta)$, which means that N increases as g decreases. Increasing N increases resolution of images, but the optimization problem for optimal binary code becomes lengthy. On the other hand, increasing Δ can increase spatial resolution, but the spectral resolution reduces. Keeping practical considerations in mind, we set $N = 32$, which took close to a day to optimize. Further, $g = 300 groves/mm$ was the smallest grove density we obtained as off-the-shelf component.



(**a**) Image to be displayed (**b**) Positive half (**c**) Negative half

(**d**) Positive and negative data (**e**) Positive measurement (**f**) Negative measurement

Figure 3.17: **Capturing data with positive and negative parts**. Since measurements are linear, we split the image to be displayed (**a**) into positive part in (**b**), with a maximum value of 0.0026 and negative part in (**c**) part, with a maximum value of 0.0029. By capturing positive (**e**) and negative data (**f**), $\mathbf{y}_p$ and $\mathbf{y}_n$ respectively, the required measurement is evaluated as $\mathbf{y} = 0.0026\mathbf{y}_p - 0.0029\mathbf{y}_n$.

3.6.2 Handling positive/negative data

When computing singular vectors, the data to be measured, as well as the data to be displayed on the LCoS contains negative values. Since our optical devices cannot handle negative data, we make two positive measurements and combine them. We split the data to be displayed on the LCoS into positive and negative parts. Then, we capture positive data with positive part on the LCoS, and then repeat the process for negative data. By taking the difference of the positive and negative data, we obtain the required measurement. Figure 3.17 shows an example of capture of data with positive/negative data. The data in (**a**) shows the positive/negative image to be displayed on the LCoS, which is split into positive (**b**) and negative (**c**) halves, which are separately displayed on the LCoS, to capture positive (**e**) and negative (**f**) data. The final required measurement is then obtained by appropriately weighing and subtracting the two measurements.

3.6.3 Calibrating the optical setup

We now outline calibration steps for the proposed optical setup. Firstly, we need a mapping between the captured image and the image displayed on the SLM. Secondly, we need calibration of wavelengths, and finally, we need spectral response calibration of the system for high-fidelity measurements.

Camera-SLM calibration. Recall that the power method for estimating eigen vectors requires the multiplication $\mathbf{x}_2 = H\mathbf{x}_1$, where $\mathbf{x}_1 = H^\top \mathbf{x}_0$ is a spatial measurement, displayed on the SLM and $\mathbf{x}_2$ is the measurement made by the camera. Hence, we need a one-to-one mapping between the measured

image and the LCoS. To do this, we added a second, calibration camera, henceforth called the auxiliary camera, which directly sees the image on the LCoS. The calibration steps are:

1. Find pixel to pixel correspondence between LCoS and auxiliary camera using gray or binary codes.
2. Place known target in front of the camera.
3. Capture the image of the target using the primary camera. Let this image be I_1.
4. Capture the image of the target on the LCoS using second camera. Let this image be I_2.
5. Register I_1 and I_2 using a similarity transform.

The steps are then repeated for the spectrum. Instead of placing a known target image, a narrow band filter is placed. This creates the coded aperture pattern on both the cameras. The image of the coded aperture for the narrow band filters can be used for registering the cameras for spectral measurements. For robustness, we combined images of two narrow band filters, namely 514.5nm with an FWHM of 1nm and 670nm with an FWHM of 3nm, which helped registration of the camera and LCoS over a larger field of view.

Figure 3.18 shows spatial and spectral calibration results. (**a**) shows the images of target captured by auxiliary camera and (**b**) shows capture by measurement camera. The calibration process was verified by displaying the captured target image back on the SLM and then capturing the image of LCoS by auxiliary camera. The result is shown in (**c**). (**d**) shows the result if the registration were not successful, showing ghosting of the two images. (**e**) and (**f**) show image of spectrum of a narrowband filter. Since the pupil code is vertically symmetric, we stuck a piece of tape at the bottom, creating a trapezoidal shape, which was then easy to register. (**g**) shows the overlay image captured by the auxiliary camera, for verification. A good registration results in an image that looks like the aperture code itself. (**h**) shows the result of an intentional shift, to show the effect of a bad registration. In both cases, we used Matlab's built in SURF based automatic image registration technique for estimating a similarity transform between the two captured images.

Wavelength calibration. Wavelength calibration requires two steps – 1) Estimating the binary code of the coded aperture and 2) Estimating locations of wavelengths. We found thresholding the measured spectrum to be a robust way of estimating the binary code of the coded aperture. To calibrate wavelength locations, we use three filters of known spectral response. Specifically, we use 488nm, 514.5nm and 670nm spectral filters with FWHM of 1nm, 1nm and 3nm respectively. Since spectral spread is linear,

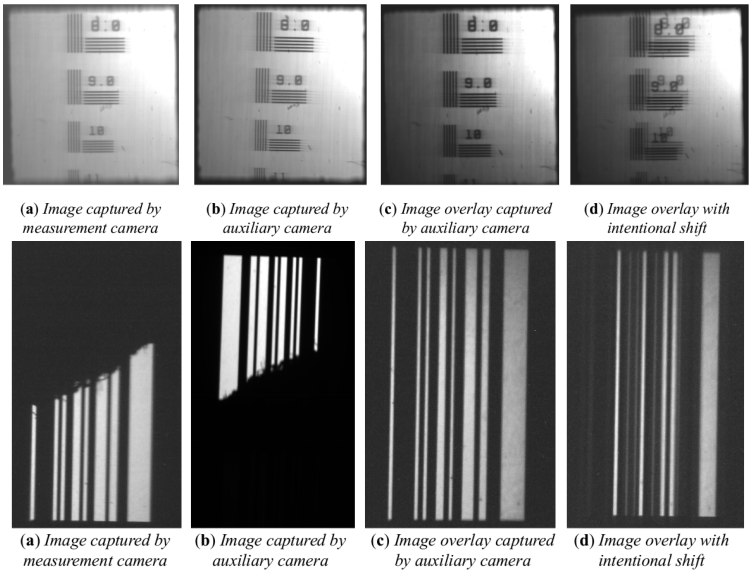
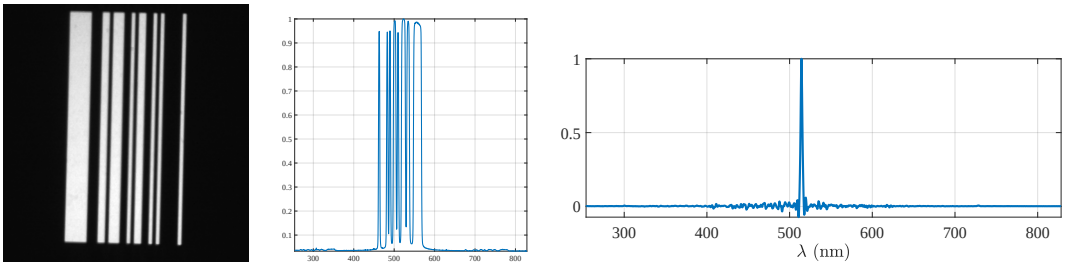


Figure 3.18: **Camera-SLM calibration for lab prototype**. Images showing spatial and spectral calibration accuracy obtained by placing a target image in front of the camera. Spatial registration is done by capturing image of a known target placed in front of the optical setup by the measurement camera (**a**) and auxiliary camera (**b**). We then register the two images to obtain a similarity transformation that maps images from the measurement camera to the LCoS. To verify our registration, we keep the target in front of the setup and display the image captured by the measurement camera on LCoS after mapping. (**c**) shows the image then captured by the auxiliary camera. The images overlap well, implying that the registration process was successful. (**d**) shows an intentional shift induced in the measured image and displayed back on the LCoS. There is a visible shift in the target image. A similar process is followed for the spectral measurement registration as well. Instead of a target, we place a narrowband filter in front of the optical setup and illuminate it with a broadband light source. Since the pupil code image is vertically symmetric, there will be a 180 degrees ambiguity. We get rid of that by sticking a tape at the bottom and capturing images, shown in (**e**) and (**f**). A successful registration process results in (**g**), with image on LCoS very well overlapping with the mapped measurement image. (**h**) shows overlay with an intentional shift, resulting in an image that does not look like the pupil code.

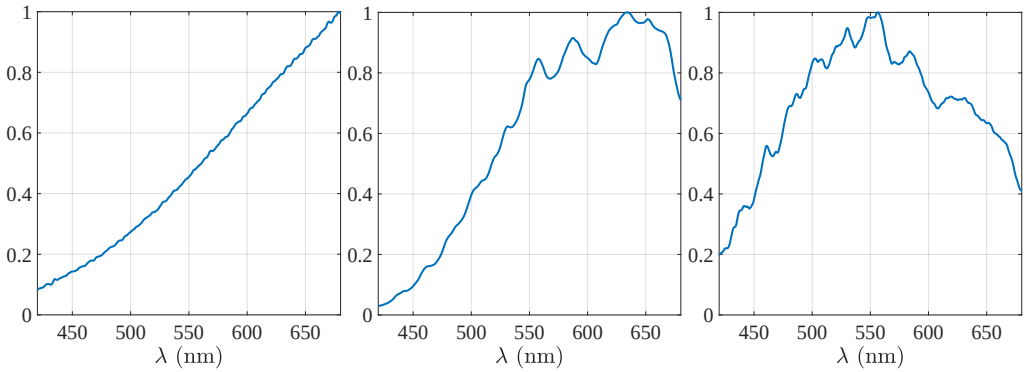
two known wavelengths are sufficient. However, for robustness, we use a third filter and then linearly interpolate to get the wavelength positions.

Figure 3.19 shows the image for wavelength calibration pipeline. We first obtain image of spectrum of a narrow band filter. After correcting for rotation, we obtain spectrum by summing the image vertically. This helps estimate the binary code, which is then used to deconvolve the observed spectrum to get spectrum of the narrow band filter. The peak of the narrow band filter is used as a known location. The



(a) Captured image **(b)** Spectrum computed from image **(c)** Deconvolved spectrum

Figure 3.19: **Calibrating of code and wavelengths location**. We start with image of spectrum of a narrow band filter (a), 514.4nm in this case. Then the image is corrected for rotation and summed vertically to obtain the spectrum (b). The spectrum is thresholded to get the binary code which is then used to deconvolve the observed spectrum to obtain the spectrum of the narrow band filter in (c). We used Wiener deconvolution for obtaining the sharp spectrum.



(a) Ground truth spectrum **(b)** Raw spectrum **(c)** System spectral response

Figure 3.20: **Calibration of spectral response**. We used the tungsten-halogen light souce “SL1-CAL" from Stellarnet. The ground truth spectrum in (a) was provided as part of the light source. We then measured spectrum of the light source by reflecting it off spectralon and deconvolving it with aperture code, shown in (b). Measured spectrum was then divided by ground truth spectrum to obtain system response, shown in (c).

process is then repeated for 488nm and 670nm filters to get wavelengths.

Spectral response of camera / Radiometric calibration. The measured image and specturm on the camera plane is given by

$$I_S(x,y) \propto \int_\lambda \left(H(x,y,\lambda) * \left| \frac{1}{\lambda f} A\left(-\frac{x}{\lambda f}, -\frac{x}{\lambda f}\right)\right|^2 \right) c(\lambda) d\lambda$$
$$I_R(x,y) \propto a(x,y) * \left(s\left(\frac{x}{\lambda f \nu_0}\right) c\left(\frac{x}{\lambda f \nu_0}\right)\right),$$

where $c(\lambda)$ is the spectral response of the camera. For true spectrometric readings, contribution of $c(\lambda)$ needs to be removed. This can be achieved by calibrating the spectrometric measurements with a known light source. The tungsten-halogen light source, "SL1-CAL" from Stellarnet was used for this purpose. To compute $c(\lambda)$, we assumed that the true spectrum, $c_t(\lambda)$ of the light source is known. We then measured spectrum of the light source, $c_m(\lambda)$ with our optical setup. The spectral response of the system was then computed as $c(\lambda) = \frac{c_m(\lambda)}{c_t(\lambda)}$. This procedure is illustrated in Figure 3.20.

Diffraction due to LCoS pattern. Since the SLM is placed $2f$ away from spectral or spatial measurements, the displayed pattern introduces diffraction blur, creating a non-linear measurement system. To counter this, we add a constant offset to both positive and negative patterns, which makes the diffraction blur compact enough that the non-linearities can be neglected.

Spectral deconvolution. Measurements by our optical system return spectra at each point, convolved by the aperture code. To get the true spectrum, we deconvolved the k^{th} measured singular vector using a smoothness prior. The specific objective function we used:

$$\min_{\mathbf{v}_k} \quad \frac{1}{2}\|\mathbf{y}_k - a * \mathbf{v}_k\|^2 + \eta\|\nabla \mathbf{v}_k\|^2, \tag{3.15}$$

where $\mathbf{v}_k$ is the true spectrum, $\mathbf{y}_k$ is the measured spectrum, a is the aperture code, $\nabla\mathbf{x}$ is the first order difference of $\mathbf{x}$, and η is weight of penalty term. Solution to (3.15) was computed using conjugate gradient descent. Higher η favors smoother spectra, and hence is preferred for illuminants with smooth spectra, such as tungsten-halogen bulb or white LED. For peaky spectra such as CFL, a lower value of η is preferred. In our experiments, we found $\eta = 1$ to be appropriate for peaky spectra, whereas, $\eta = 10^3$ was appropriate for experiments with tungsten-halogen illumination. We compare performance of various deconvolution algorithms in the supplementary section.

Spatial deconvolution. Equation (3.7) suggests that the spatial blur kernel varies across different spectral bands. More specifically, the blur kernels at two different spectra are scaled versions of each

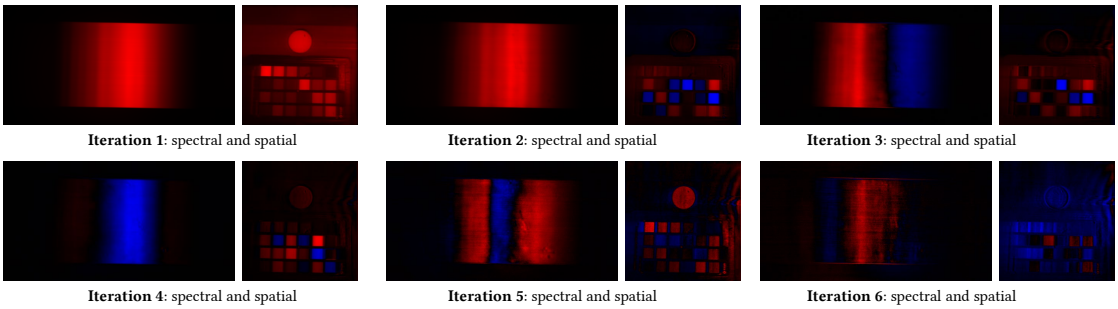


Figure 3.21: **Visulaization of Krylov iterations**. Data captured during measurement process for a rank-4 approximation of the "Color checker" scene for six iterations. A picture of the scene is shown in Figure 3.24. Positive part of data is shown in red and the negative part is shown in blue. KRISM alternates between acquiring spectral and spatial measurements to compute both spatial and spectral singular vectors. The first four iterations involve capturing the dominant wavelengths that includes yellow and green colors, since they have the highest magnitude. The next set of iterations capture the blue and red wavelengths.

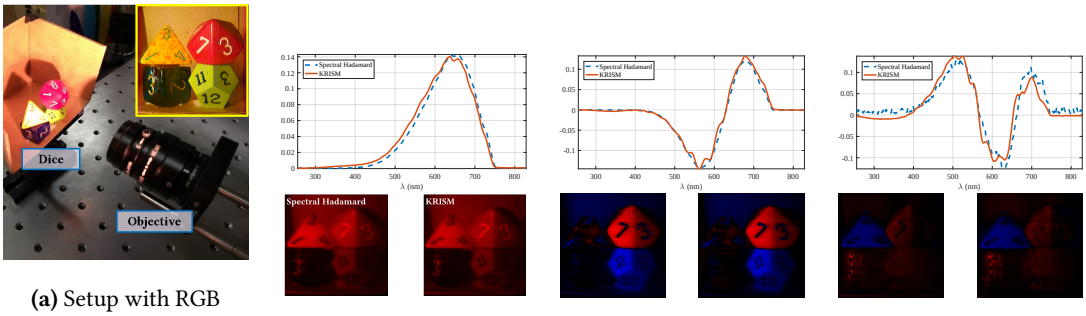
other. However, we observed that the variations in blur kernels were not significant when we image over a small waveband — for example, the visible waveband of $420-680$nm. Given this, we approximate the spatial blur as being spectrally independent, which leads to the following expression:

$$I_S(x,y) \propto \left[\int_\lambda H(x,y,\lambda)c(\lambda)d\lambda\right] * p(x,y), \tag{3.16}$$

where $p(x,y)$ is the spatial blur. We estimated the spatial blur kernel by imaging a pinhole and subsequently deconvolved the spatial singular vectors. We used a TV prior based deconvolution using the technique in [12] using the image of a pinhole as the PSF. Details of the deconvolution procedure are in the supplementary section.

3.7 Real Experiments

We present several results from real experiments which show the effectiveness of KRISM. We evaluate the ability to measure singular vectors with high accuracy, and high spatial and spectral resolution capabilities. Unless specified, experiments involved a capture of a rank-4 approximation of the HSI, with 6 spectral and 6 spatial measurements. Lanczos iterations were initialized with all-ones spatial image to speed up convergence. HSIs were acquired with a spatial resolution of 560×550 pixels and a spectral resolution of 256 bands between 400nm to 700nm, with 3 nm FWHM. For verifying spectroradiometric capabilities, we obtained spectral measurements at a small set of spatial points using an Ocean Optics FLAME spectrometer. We use spectral angular mapper (SAM) [103] similarity and PSNR between spectra



(a) Setup with RGB image in inset

(b) First singular vector

(c) Second singular vector

(d) Third singular vector

Figure 3.22: **Comparison of captured singular vectors**. The left image singular vector is from Hadamard multiplexed data and the right one is from KRISM. Blue represents negative values and red represents positive values. KRISM required a total of 6 spectral and 6 spatial measurements to construct 4 singular vectors. While spectral Hadamard sampling method took 49 minutes, KRISM took under 2 minutes. The SAM value between the singular vectors was less than 20°.

measured by our optical setup and that measured with a spectrometer. SAM between two vectors $\mathbf{x}$ and $\widehat{\mathbf{x}}$ is defined as $SAM = \cos^{-1}\left(\frac{\mathbf{x}^\top \widehat{\mathbf{x}}}{\|\mathbf{x}\|\|\widehat{\mathbf{x}}\|}\right)$.

Visualization of Lanczos iterations Figure 3.21 shows iterations for the “Color checker" scene in Figure 3.24. The algorithm initially captures brightest parts of the image, corresponding to the spectralon, and the white and yellow patches. Consequently, by iteration 5, the blue and red parts of the image are isolated. The iterations are representative of the signal energy in various wavelengths. Maximum energy is concentrated in yellow wavelengths, due to tungsten-halogen illuminant and spectral response of the camera. This is then followed by the red wavelengths, and finally the blue wavelengths.

Comparison of measured singular vectors. We obtain the complete hyperspectral image through a permuted Hadamard multiplexed sampling in the spectral domain for comparison with ground-truth singular vectors. We chose a scene with four colored dice for this purpose, shown in Figure 3.22 (**a**). We then computed 4 singular vectors of spectrally Hadamard-multiplexed data. Figure 3.22 shows a comparison of the spatial and spectral singular vectors. The singular vectors obtained via Krylov subspace technique are close to the ones obtained through Hadamard sampling. On an average, the reconstruction accuracy between KRISM and Hadamard multiplexing was found to be greater than 30dB, while the angle between the singular vectors was no worse than 20°, with the top three singular vector having

an error smaller than 8°. Hadamard sampling took 49 minutes while KRISM took under 2 minutes for 6 spatial and 6 spectral measurements, thus offering a speedup of 20×.

Peaky spectrum illumination We imaged a small toy figurine of “Chopper", placed under CFL, which has a peaky spectrum, to test high spatio-spectral resolving capability. Figure 3.1 shows the rendered RGB image and spectra at a representative location. Spectra at a selected spatial point, as measured by KRISM, and a spectrometer are shown as well. The SAM between spectrum measured by KRISM and that measured by spectrometer was found to be 14.7°. Notice that the location of the peaks, as well as the heights match accurately. Indeed, the chopper example establishes the high spatio-spectral resolution capabilities of KRISM.

Diverse real experiments. Figure 3.23 shows several real world examples captured with our optical setup, with a diverse set of objects. For verification with ground truth, we captured spectral profiles at select spatial locations. The “Dice" and “Objects" scene captures several more colorful objects with high texture. The zoomed-in pictures show the spatial resolution, while the comparison of spectra highlights the fidelity of our system as a spectral measurement tool. “Ace" scene was captured by placing the toy figurine under CFL illuminant, which is peaky. We could not obtain ground truth with a spectrometer, as the toy was too small to reliably probe with a spectrometer. The peaks are located within 2nm of ground truth peaks, and the relative heights of the peaks match the underlying color. “Crayons" scene consists of numerous colorful wax crayons illuminated with a tungsten-halogen lamp. The closeness of spectra with respect to spectrometer readings shows the spectral performance of our setup. Finally, “Feathers" consists of several colorful feathers illuminated by tungsten-halogen lamp. The fine structure of feathers is well captured by our setup.

Color checker. Since our setup is optimized for viewing in 400nm-700nm, we evaluated our system on the 24-color Macbeth color chart. The Macbeth color chart consists of a wide gamut of colors in visible spectrum that are spectrally well separated, and forms a good test bench for visible spectrometry. We placed the “Color passport" and spectralon plug in front of our camera and illuminated it with a tungsten-halogen DC light source. The spectralon has a spectrally flat response, and hence helps estimate the spectral response of the illuminant+spectrometer system. This enables measurement of true radiance of the color swatches. Since the spectra is smooth, we used least squares recovery of the spectrum, with ℓ_2 penalty on the first difference of spectral singular vectors. The captured data was then normalized by dividing spectrum of all points with the spectrum of the spectralon. Figure 3.24 shows the captured

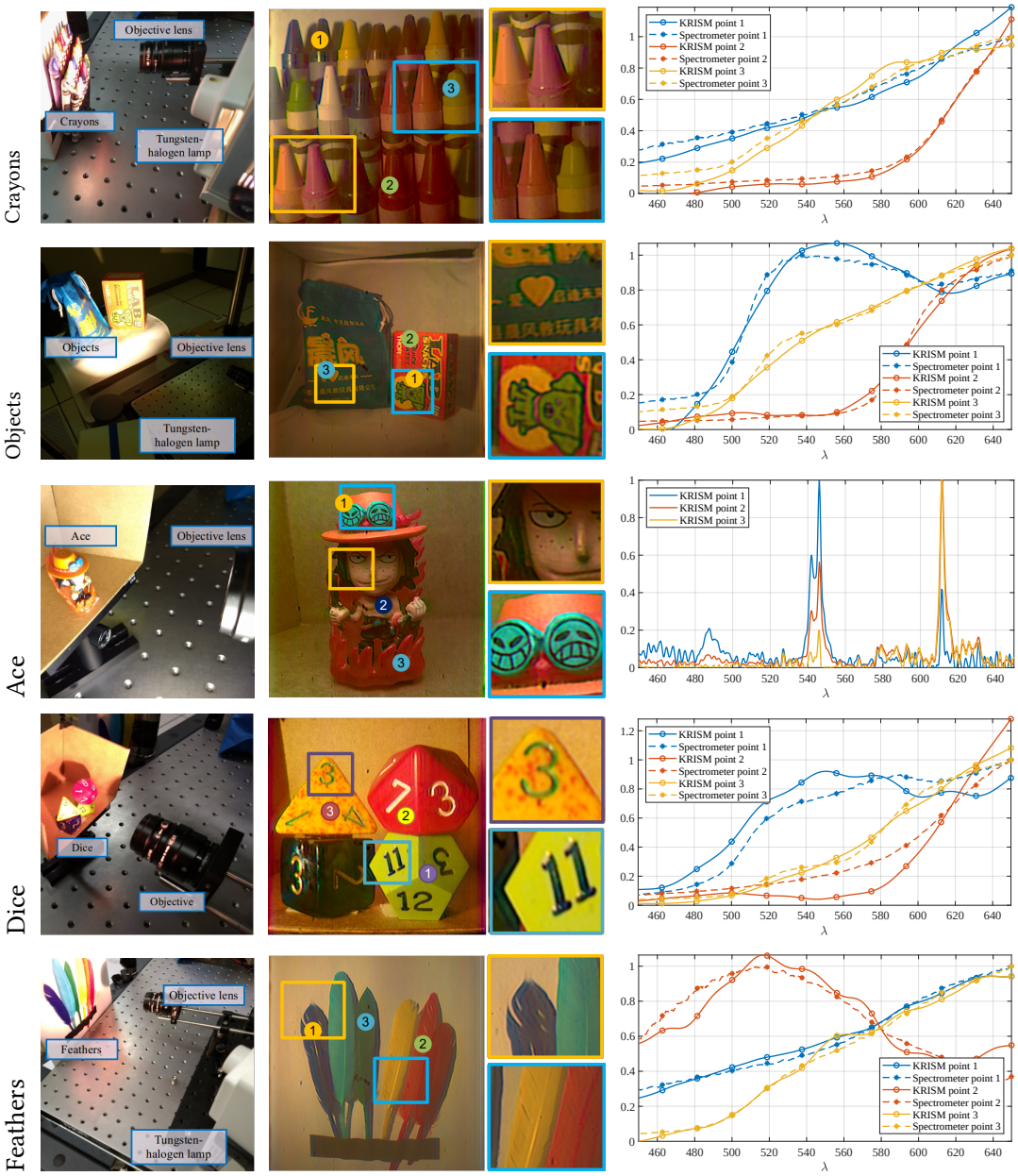
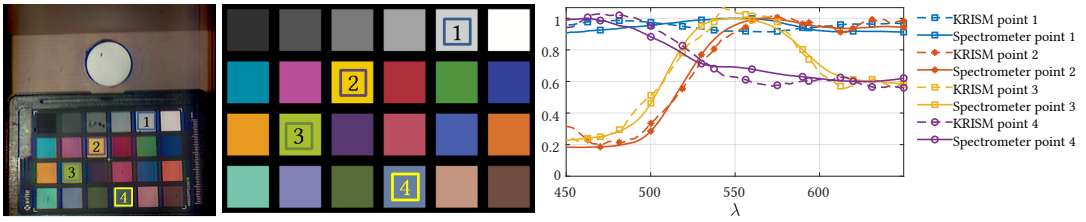


Figure 3.23: **Real data captured with our optical setup**. We show the physical setup used for capturing the data, rendered RGB image with some interesting patches zoomed in, and spectra at some points, compared with a spectrometer. The results are promising, as the spectra is very close to spectrometer readings (PSNR > 20dB), and the spatial images are captured in high resolution.



(a) RGB image **(b)** Reference RGB image **(c)** Spectral profiles

Figure 3.24: **Macbeth color chart**. Spectra is shown at four locations and compared with spectrometer readings. The PSNR is 25dB or higher and the SAM between KRISM spectra and spectrometer readings is less than 6°.

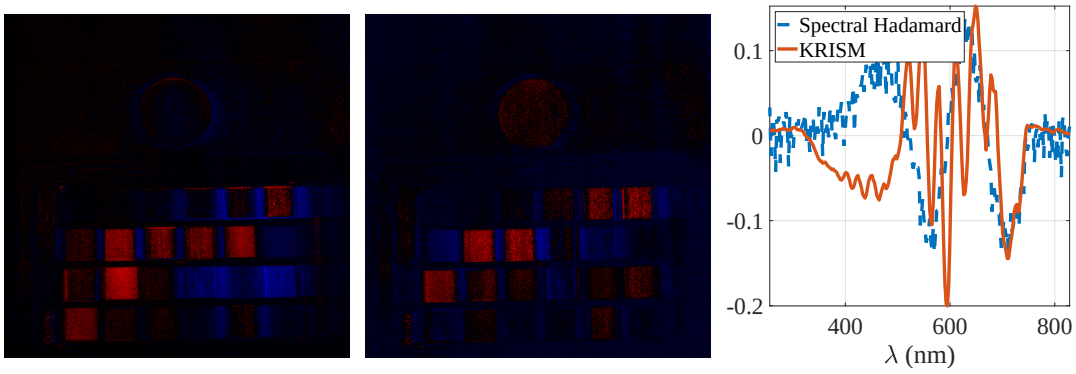
image against reference color chart along with spectra at select locations plotted along with ground truth spectra. On an average, the PSNR between spectra measured by KRISM and that measured by spectrometer is greater than 25dB, while the SAM is less than 6°.

3.8 Discussion and conclusion

We presented a novel hyperspectral imaging methodology called KRSIM, and provided an associated novel optical system for enabling optical computation of hyperspectral scenes to acquire the top few singular vectors in a fast and efficient manner. Through several real experiments, we establish the strength of KRISM in three important aspects: 1) the ability to capture singular vectors of the hyperspectral image with high fidelity, 2) the ability to capture an approximation of the hyperspectral image with 20× or faster acquisition rate compared to Nyquist sampling, and 3) the ability to measure simultaneously at high spatial and spectral resolution. We believe that our setup will trigger several new experiments in adaptive imaging for fast and high resolution hyperspectral imaging.

Added advantages. There are two additional advantages to KRISM. One, since we capture the top few singular vectors directly, there is a data compression from the acquisition itself. Two, the only recovery time involves deconvolution of a few spatial and spectral singular vectors, which is significantly less than the time required for recovery of hyperspectral images from CS measurements.

Beyond low-rank volumes. Key to our method is the assumption that the underlying HSI is low-rank. Sensing a high rank HSI will require several measurements which negates the benefits of KRISM. However, there are several other matrix sampling techniques that rely on row or column sensing [41, 71] to capture information about high rank matrices in an efficient manner. Since the proposed setup is capa-



(a) Spectral Hadamard **(b)** KRISM **(c)** Spectra

Figure 3.25: **Capturing higher singular vectors**. Since KRISM computes higher singular vectors by progressively blocking more light, photon noise dominates measurements after some iterations resulting in noisy estimates of singular vectors. The above example shows inaccurately estimated fifth singular vectors measured for “color checker" scene with our lab prototype.

ble of computing arbitrary matrix-vector products, such matrix sampling techniques can be implemented efficiently.

Effect of photon noise Although Krylov subspace based methods are very robust to noise [91], the quality of the singular vectors degrade as the rank of acquisition is increased (see Figure 3.25). This is primarily due to photon noise, as we progressively block most of the energy contained in initial singular vectors. This can be mitigated by increasing the exposure time of measurements for higher singular vectors. All said, the problem of noisy higher singular vectors exists with any kind of sampling scheme and hence needs separate attention via a good noise model.

4 Programmable Spectrometry – Per-pixel Material Classification using Learned Spectral Filters

We now tackle an important application of hyperrspectral imaging – per pixel material classification. We are now armed with a high-resolution spectrally-programmable camera from the previous chapter. We will use this in conjunction with learned spectral filters to perform per-pixel classification with as few images as possible. Towards the end of this chapter, we will see how to learn discriminating spectral filters from a pool of spectrometric measurements, a compact optical setup to implement spectral filtering, and methods to achieve highly accurate binary and multi-class classification results.

4.1 Introduction

Material composition of a scene can often be identified by analyzing variations of light intensity as a function of spectrum or wavelengths. Since materials tend to have unique spectral profiles, spectrum-based material classification has found widespread use in numerous scientific disciplines including molecular identification using Raman spectroscopy [20], tagging of key cellular components in fluorescence microscopy [56], land coverage and weather monitoring [19, 39], and even the study of chemical composition of stars and astronomical objects using line spectroscopy. It would not be a stretch to suggest that spectroscopy or its imaging variant, hyperspectral imaging (HSI), is an important scientific tool for material identification.

While hyperspectral imaging has also found application in computer vision tasks [48, 72, 95], its widespread adoption has been hindered due to inherent challenges in their acquisition. Measuring a HSI requires sampling of a very high dimensional signal; for example, mega-pixel images at hundreds of spectral bands, a process that is daunting to do at video rate. This problem is further aggravated by the fact that hyperspectral measurements have to combat low signal to noise ratios, as a fixed amount of light is divided in to several spectral bands — leading to long exposure times that can even span several minutes per HSI.

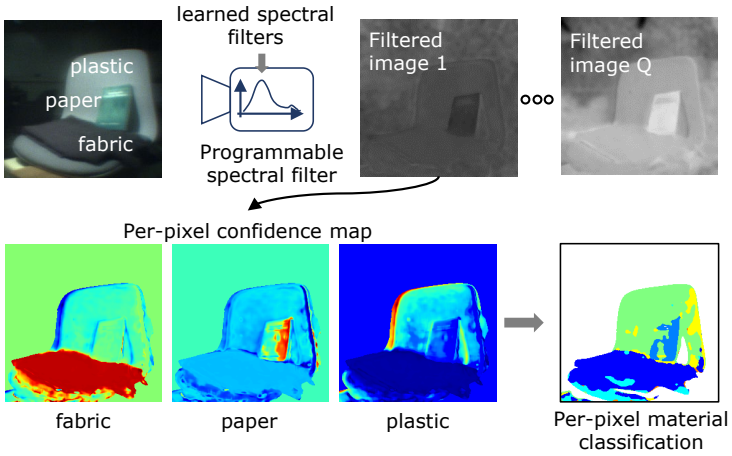


Figure 4.1: **Spectrum-based material classifier.** We propose an optical setup that is capable of classifying material on a per-pixel basis. This is achieved by capturing a small number of spectrally filtered images of the scene and then performing classification. This results in a system that requires far fewer measurements than fully scanning the hyperspectral image and then classifying, and leads to a high light throughput system, as the discriminant spectral filters tend to be broadband.

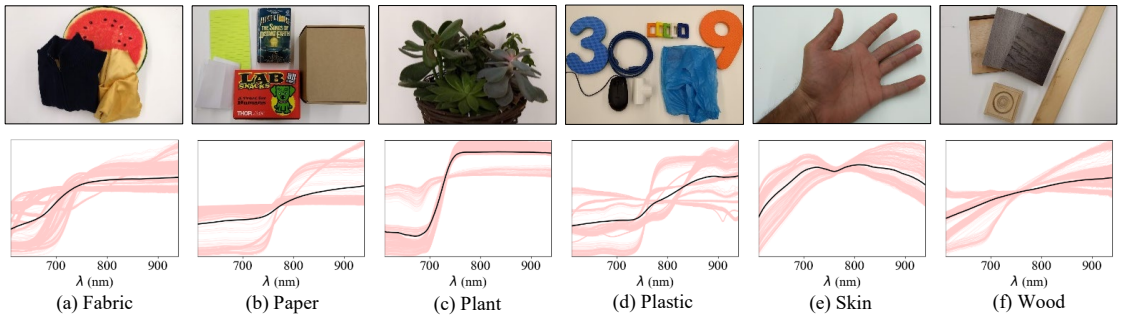


Figure 4.2: **Spectrum of materials.** Spectrum is often a unique identifier of materials and can be used for per-pixel classification tasks. We collected spectral profiles of several everyday materials and have shown images of some of the objects we sampled as well as the spectral profiles, with the black line indicating the average profile for each class. We used the collected spectral profiles to train spectral filters that enable high accuracy, per-pixel material classification.

This chapter proposes a novel approach for enabling spectrometry-based per-pixel material classification by overcoming the limitations posed by HSI acquisition. To understand our proposed approach, we first need to delve deeper into the process of classification itself. Classification techniques involve comparing the spectral profile at each pixel with known or *learned* spectra by taking a linear projection. Intuitively, given K material classes, we would compute $O(K)$ such linear projections. For example,

a support vector machine (SVM) classifies by finding distance of features from the separating hyperplane; in the context of spectral classification, this translates to spectrally filtering the scene with the hyperplane coefficients. Hence, spectral classification can be made practical if we can capture the linear projections directly without having to acquire the complete HSI. Such an operation translates to optically filtering the scene's HSI using known spectral filters, which can be achieved if the camera's spectral response can be arbitrarily programmed.

To enable per-pixel material classification, we propose a new imaging architecture with a programmable spectral response that can be changed on-the-fly at video rate. Given a training dataset of spectral profiles, we use off-the-shelf classification techniques like SVMs and deep neural networks to identify linear projections that facilitate material classification. For a novel scene, the camera captures multiple images, each with a different spectral response; the captured measurements are used with the classifier to perform per-pixel material classification.

We propose a per-pixel material classification camera and make the following contributions:

1. *Optical computing for classification.* By optically evaluating the linear projections, we circumvent sampling the full HSI. This requires significantly fewer measurements (less than 10 vs. several hundreds), and has higher light efficiency due to broadband spectral filters.

2. *Simple strategy for learning spectral filters.* Since material classification only requires spectral profiles, we show that discriminant filters can be learned on labeled spectral measurements. This simplifies the dataset problem, as collecting data with a spectrometer is much simpler than capturing a full HSI and then labeling objects. As part of this thesis, we collected several spectral profiles of everyday objects and learned discriminant filters (see Fig. 4.2). We aim to release our dataset to facilitate future research in spectrum-based material classification.

We demonstrate our proposed method with a lab-built spectrally-programmable camera that is capable of imaging at high spatial resolution while filtering at high spectral resolution. For binary classification problem, our lab prototype provides a classification result every alternate frame, and achieves a processing rate of 15 frames per second (half of maximum). We also show results on multi-class labeling problems using a classifier that can differentiate between six distinct material types.

4.2 Prior Work

We discuss prior work in the areas of material classification using HSIs as well as optical computing and design of programmable spectral filters.

4.2.1 Spectral Classification

Given spectral profile measurement, $s(\lambda)$, which is often obtained using a spectrometer, the goal of spectral classification is to estimate the material composition of the object. There are several classical techniques such as support vector machines (SVM) that use $s(\lambda)$ as a feature vector. In more recent times, neural networks have been used to classify every day materials with spectra of objects measured with a handheld spectrometer [25]. This process is often computationally light weight – as spectrum is a unique identifier of materials. Its extension to classification of all pixels in the image of scene is the next step, but a very challenging task. Classification at image level requires capturing spectrum at each pixel – thereby requiring a full scan of the hyperspectral cube which is often daunting. We will next see how hyperspectral classification is performed, which will then motivate the proposed *optical classification* architecture.

4.2.2 Hyperspectral classification

Consider the HSI of a scene, $H(x, y, \lambda)$, where each pixel (x, y) is assumed to belong to one of K material classes. Specifically, the spectra at each pixel can be written as,

$$H(x, y, \lambda) = \alpha(x, y) S_{L(x,y)}(\lambda), \tag{4.1}$$

where $L(x, y)$ is label of the material contributing to spectrum at (x, y), and $\alpha(x, y)$ is scaling parameter. Note that the model above assumes all spatial pixels are pure, i.e., every pixel gets contribution from only one material. We use this model for the sake of exposition and later discuss about how to relax it later to handle mixed pixel.

The goal of classification is to estimate the label at each pixel, $L(x, y)$, which forms a label map. There are broadly two approaches to spectral classification — generative and discriminative. Generative techniques rely on decomposing the HSI as a linear combination of basic materials that are called end-members [24]. Specifically, the HSI of the scene is decomposed as,

$$H(x, y, \lambda) = \sum_{k=1}^{K} s_k(\lambda) a_k(x, y), \tag{4.2}$$

where $s_k(\lambda)$ is the spectra of k^{th} material, and $a(x, y)$ is the relative contribution of material k at (x, y). The abundances at each pixel along with the end-member spectra provide a feature vector that can be used to spatially cluster the materials and subsequently identify them.

Discriminative techniques rely on directly learning discerning features from the HSI without the intermittent stage of low-dimensional decomposition. Here, we identify a set of spectral filters,

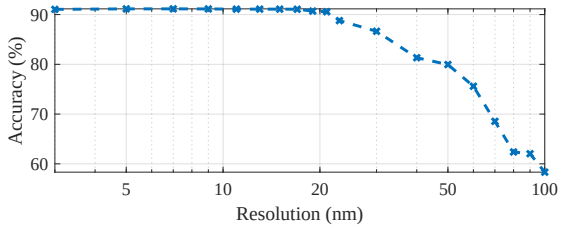


Figure 4.3: **Effect of resolution on classification accuracy**. In order to classify accurately, it is important to sample the spectral profiles at high resolution. The plot shows classification as a function of spectral resolution for a 6-class classification task with a one-vs-rest SVM classifier. High sampling rates ensure high accuracy; however, such high spectral resolutions invariably require long exposure durations, making them infeasible for dynamic scenes. Our approach is to circumvent measuring the full HSI and perform the necessary computations for classification in the optical domain.

$\{(d_k(\lambda), \beta_k)\}_{k=1}^{M}$ that generate per-pixel feature vector via spectral-domain filtering:

$$F_k(x, y) = \int_{\lambda} H(x, y, \lambda) d_k(\lambda) d\lambda + \beta_k. \tag{4.3}$$

Hence, each image $F_k(x, y)$ is a spectrally-filtered version of the HSI with an added offset. In case of SVMs, the learned spectral filters form separating hyperplanes; this has been a *de facto* way of HSI classification [26, 65]. More sophisticated learning techniques based on neural networks use spectral features [44] or spatio-spectral features [17, 38, 42, 51, 55, 60, 64, 90] for classification.

4.2.3 Need for high spectral resolution

A key requirement for highly accurate classification is high spectral resolution. Sampling at as fine as 1nm helps discriminate between materials very easily, while very low sampling resolution, such as an RGB camera that samples across several 100 nm, is incapable of detecting materials accurately. To drive home this message, we performed a 6-class classification with spectrometric data measured at varying resolutions. A glimpse of the data is presented in Fig. 4.2 and we will elaborate on it later. We built a simple, one-vs-rest (SVM) classifier, and estimated testing accuracy for varying spectral resolution, starting from 3nm down to 100nm. Figure 4.3 shows the 6-class classification accuracy. There is a clear reduction in accuracy as resolution goes down – indicating the necessity for high spectral resolution. However, completely scanning the spectral profile and then classifying is a wasteful process. Invariably, the number of spectral features used, i.e, the dimensionality of the projection, tends to be smaller than the number of spectral channels in the HSI. Hence, we seek to measure the features directly, by com-

puting (4.3) optically. As is to expected, such a paradigm of *optical classification* requires the design of programmable cameras capable of implementing an arbitrary spectral filter.

4.2.4 Sensing for classification

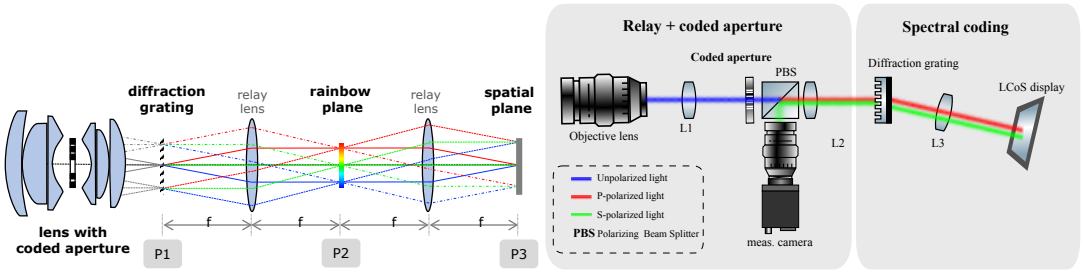
There are several techniques that rely on capturing compressively sensed (CS) measurements with the aim of classification. Li et al. [54] perform this by capturing a small set of randomly spatially-multiplexed measurements and then recovering labels via an unmixing framework. Ramirez et al. [79–81] implemented a spatial-spectral multiplexed imager that captures a single snapshot with a spatial coded aperture and then performs classification. While such methods reduce the hardware complexity, much of the burden is placed on the classifier itself. Specifically, the measurement stage (linear, random) is decoupled from the classification stage (unmixing). However, since most classification strategies involve a linear transformation of the spectral profile, we can move part of the classification computations to optical domain.

4.2.5 Optical computing

Instead of relying on both spatial and spectral information, we consider a simpler approach which relies only on the spectral profiles for classification. Such a strategy is less accurate than spatial and spectral versions [17, 38, 42, 51, 55, 60, 64, 90], but significantly reduces the complexity of the imaging system. This approach is similar, in spirit, to using BRDFs to perform per-pixel classification by varying the incident illumination [37, 61], or using first layer of a neural network to capture light fields [16]. Such a setup requires far fewer measurements, as the number of material classes is significantly smaller than the number of spectral bands. Further, since the linear projections are computed optically, the contribution of readout noise is minimized. Optical computing has found use in various computer vision tasks such as capturing light transport matrices [70], low-rank approximation of hyperspectral images [87], and spectral classification using programmable light sources [32, 73]. We adopt the paradigm of optical computing to make discriminative filter measurements by building a camera whose spectral response can be arbitrarily programmed.

4.2.6 Dynamic spectral filters

Spectral filtering can be achieved by modified the response of the camera; a canonical and static example being the Bayer pattern or more interestingly, the case of fluorescence filters in microscopy. It is however more useful to have a camera whose response can be altered arbitrarily in a fast manner. Numerous



(a) Setup schematic **(b)** Practical realization

Figure 4.4: **Schematic for programmable spectral filter.** The optical architecture in (a) consists of a lens assembly with coded aperture which introduces spatial and spectral blurs. By placing an SLM in P2, the HSI of the scene can be spectrally filtered and sensed by a camera sensor on P3. (b) shows a compact realization of the setup.

techniques to achieve spectral filtering have been proposed in the past. Agile spectral imager [66] rely on the coding the so-called “rainbow plane" to achieve arbitrary spectral filtering. This was further developed by [63] where they placed a digital micromirror device (DMD) on the rainbow plane to achieve dynamic spectral filtering.

However, such architectures come with a debilitating problem — usage of simple pupil codes such as open aperture or a slit directly tradeoff spatial resolution for spectral resolution. This was identified in [87] in the context of hyperspectral imaging. They showed that a slit, a common choice for spectrometry, leads to large spatial blur. Similarly an open aperture, a common choice for high-resolution imaging, leads to large spectral blur. Hence, such apertures are not conducive for high-resolution spectral classification.

We instead rely on the optical setup in [87] to overcome the spatial-spectral tradeoff. The key idea is to use a coded aperture that introduces an invertible blur in both spatial and spectral domains. An important difference is that the setup in the previous chapter is designed for HSI image acquisition; this chapter adapts the underlying ideas for performing per-pixel material classification in the scene.

4.3 Programmable Spectral Filter

Our optical setup is a modification of the optical setup proposed in [87]. We explain the relevant parts of the optical setup here.

4.3.1 4f system for spectral programming

We borrow the optical schematic for spectral programming from [87], shown in Fig. 4.4(a). Given the HSI, $H(x, y, \lambda)$, that is focused on the grating at P1, we seek to derive the intensity on planes P2 and P3. The intensity on rainbow plane P2,

$$I_4(x, y) = a^2(-x, -y) * \left(S\left(\frac{x}{f\nu_0}\right) \widetilde{c}\left(\frac{x}{f\nu_0}\right)\right), \tag{4.4}$$

where $S(\lambda) = \int_{(x,y)} H(x, y, \lambda)$ is spectrum of the scene, $\widetilde{c}(\lambda)$ is response of the optical system, and ν_0 is the density of groves in mm^{-1}. The intensity on image plane P3,

$$I_5(x, y) = \int_\lambda \left(H(x, y, \lambda) * \left| \frac{1}{\lambda^2 f^2} A\left(-\frac{x}{\lambda f}, -\frac{y}{\lambda f}\right)\right|^2 \right) d\lambda, \tag{4.5}$$

where $A(u, v)$ is the 2D Fourier transform of $a(x, y)$. The key observation from (4.4), (4.5) is that a coded aperture placed on plane P2 causes a spectral blur given by $a(x, y)$ and a spatial blur given by $\left|A\left(-\frac{x}{\lambda f}, -\frac{y}{\lambda f}\right)\right|^2$. As we saw in previoius chapters, a slit causes a severe spatial blur, whereas an open aperture causes large spectral blur. The solution is to introduce an invertible blur in both domains, which can be achieved using a coded aperture, shown in the last column. We use the same coded aperture that was used in [87], as it is designed to promote invertibility in both domains.

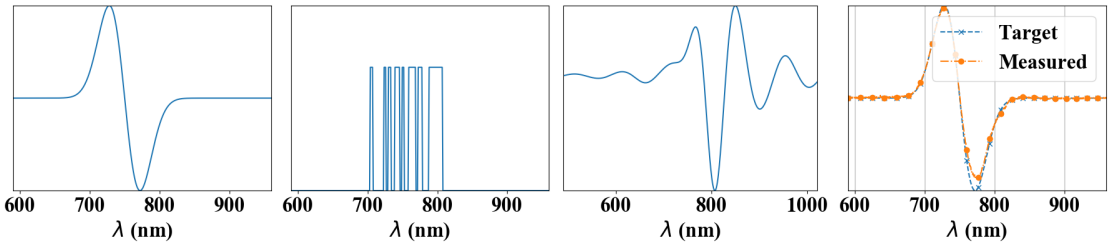
4.3.2 Optical setup

Our optical setups is in principle similar to Fig. 4.4(a). We place a spatial light modulator on the rainbow plane (P2) and sensor on spatial plane (P3) to achieve spectral filtering. The optimized binary code [87] is placed in the lens assembly Figure 4.4(b) shows a schematic of a practical implementation of the same optical setup. We use a Liquid Crystal on Silicon (LCoS) display as a spatial light modulator for spectral filtering.

4.3.3 Effect of coded aperture

Spectral filtering can be achieved by loading spatial patterns on to the rainbow plane. We will now understand what pattern to display in order to achieve a desired spectral profile $s_k(\lambda)$. For simplicity, we drop the y coordinate on all planes and only look at x-coordinate. Let $i(x_1, \lambda)$ be the complex amplitude of a point on plane P1. Let $p_k(x_2)$ be the pattern displayed on the LCoS on plane P2. Then the resultant filtering operation on a spatial point on P3 is given by,

$$\hat{i}(x_3, \lambda) = \int_{x_2} p_k(x_2) a(-x_2 + \lambda f \nu_0) i(x_1, \lambda) dx_2. \tag{4.6}$$



(a) Target spectrum $s_k(\lambda)$ (b) Coded aperture, $a(\lambda f v_0)$ (c) Display pattern, $p(\lambda f v_0)$ (d) Spectrometer measurement

Figure 4.5: **Effect of coded aperture.** To display a desired spectral filter (a) $s_k(\lambda)$, we need to project a pattern such that the convolution of the (c) projected pattern with the flipped coded aperture is equal to the target filter. (d) compares the output of our optical setup with a spectrometer and the target spectral filter.

To achieve a spectral filter with $s_k(\lambda)$, we require,

$$\int_\lambda i(x_1,\lambda)s_k(\lambda) = \int_\lambda \hat{i}(x_3,\lambda)d\lambda \tag{4.7}$$

$$= \int_\lambda \int_{x_2} p_k(x_2)a(-x_2+\lambda f v_0)i(x_1,\lambda)dx_2 \tag{4.8}$$

$$\implies \int_{x_2} p_k(x_2)a(-x_2+\lambda f v_0) = s_k(\lambda), \tag{4.9}$$

implying that we need to display a profile $p_k(x_2)$ which when *convolved* with a flipped version of the coded aperture gives rise to $s_k(\lambda)$. Since $a(x)$ was designed to be invertible, we can guarantee that there always exists a solution to eq. (4.9), and is given by,

$$p_k(x_2) = \text{deconv}\left(s_k\left(\frac{x_2}{f v_0}\right), a(-x_2)\right), \tag{4.10}$$

where “deconv" is a simple 1D deconvolution function, such as Wiener deconvolution. Figure 4.5 shows some example filters we implemented on our optical setup. We directed broadband light into the optical setup and measured the output with a spectrometer by replacing the measurement camera. We found that our setup was able to accurately create Gaussian-shaped narrowband filters of 5nm or more, thereby setting the achievable spectral resolution of the system at 5nm.

4.4 Learning Discriminant Filters

Our camera is capable of implementing any classifier that relies on a linear projection as a first step. For the sake of exposition, we illustrate two classifiers, namely support vector machine (SVM) and a deep neural network (DNN) architecture.

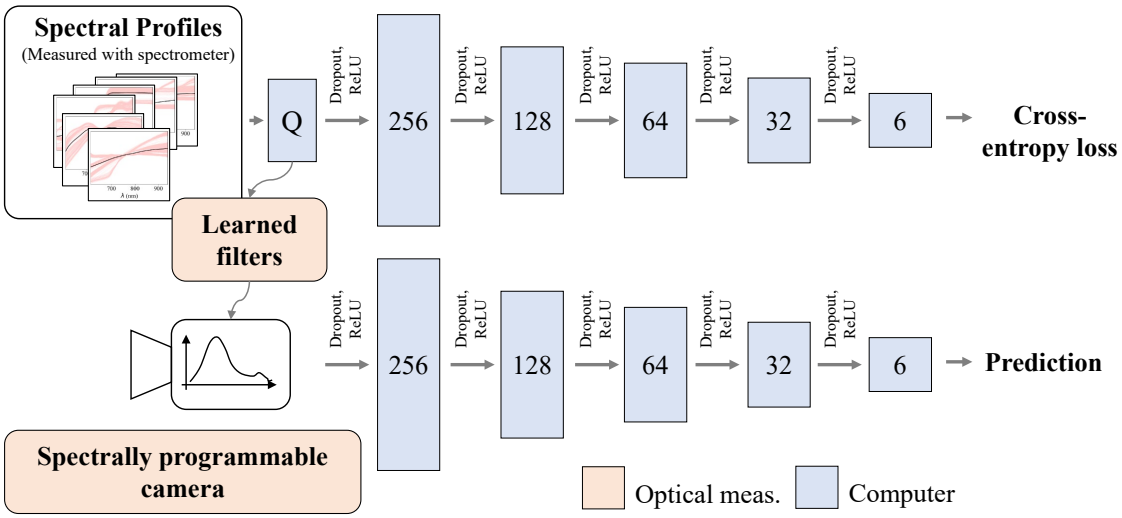


Figure 4.6: **Proposed optical classifier.** The proposed optical classifier broadly consists of two stages. In the first stage, we learn the weights of a neural network with spectrum as input and class label as output. The training process outputs the set of discerning filters, marked "learned filters" in the image. In testing stage, we filter the HSI of the scene with the learned filters, thereby replacing the first layer of the classifier with an optical implementation. This results in a high accuracy, per-pixel classifier while requiring far fewer measurements than the size of the HSI.

4.4.1 Support Vector Machine

SVMs provide a binary, linear classifier by learning a separating hyperplane on the training dataset. Given a set of data points $\{\mathbf{x}_k, y_k\}_{k=1}^{N}$, where $y_k \in \{0, 1\}$ is the label of $\mathbf{x}_k$, SVM seeks to solve the following optimization problem,

$$\min_{\mathbf{w},c} \frac{1}{N} \sum_{k=1}^{k} \max(0, 1 - y_k(\mathbf{w}^\top \mathbf{x}_k + c)) + \lambda \|\mathbf{w}\|^2, \tag{4.11}$$

where λ is a tuning parameter. The output of solving the optimization problem is the vector $\mathbf{w}$ and intercept c. In the context of optical classification, $\mathbf{w}$ is the filter that maximizes accuracy for binary decision.

4.4.2 Deep Neural Networks

Deep neural networks (DNNs) provide a richer alternative to SVMs. The architecture used in this chapter is illustrated in Fig. 4.6. We model the first linear unit of the DNN to be the programmable spectral filter. We then train a model whose input is the spectral profile at each pixel and output is the material class label as a one-hot vector. The weights of first fully connected layer are the learned discriminating filters,

Method	Classifier	Coding strategy	#Measurements	Accuracy
Santara et al.	DNN	Non-linear, spatial and spectral	220	96.7% (*reported*)
Hu et al.	DNN	Convolutional, spectrum-only	220	90.16% (*reported*)
Lee et al.	DNN	Convolutional, spatial and spectral	220	93.6% (*reported*)
Melgani et al.	SVM	**Linear, spectrum-only**	**16**	84% (*computed*)
Proposed	DNN	**Linear, spectrum-only**	**16**	90% (*Computed*)

Figure 4.7: **Simulations on the Indian Pines dataset.** We compare state-of-the-art classifiers against the classifiers proposed in this chapter. By *reported* we report the accuracy figures listed in the respective papers, while *computed* results were generated by us. A key feature of our optical setup is that it can only compute linear projections of spectra. While this leads to reduction in accuracy, the number of captured images are far fewer.

and hence the first layer can be evaluated optically, thereby circumventing the need to measure the full spectrum at each pixel. The number of filters, Q depends on the number of materials and how easily they can be separated. In our experiments, we classified a total of 6 objects. We then varied the number of filters and computed mean classification accuracy. Based on this, we picked the optimal number of filters. We note that the idea of optically computing the first layer has been explored before in the context of designing color filter arrays [14] and processing light fields [16].

4.4.3 Simulations

To evaluate accuracy of linear-only classifiers, we compare SVM and the 5-layer DNN classifier to some of the state-of-the-art techniques in spectral-classification on the NASA Indian Pine dataset which consists of 220 spectral bands with 16 object classes. We provide the relevant numbers here and provide all other details in supplementary. Figure 4.7 tabulates the accuracy of various classifiers, with the proposed methods highlighted in bold. We observe that the accuracy is lower than state-of-the-art, which is expected as we only use spectral information, while the other techniques use both spatial and spectral information. However, relying on a spectrum-only classifier lets us capture far fewer images than the number of spectral bands.

4.5 Experiments

We demonstrate capabilities of our setup for video-rate binary classification with binary SVM as well as matched filtering, and multi-class classification with multi-class SVM and DNNs.

4.5.1 Learning details

We provide details about our training process with emphasis on choice of parameters and hyperparameters. We captured a total of $1,000,000$ spectral profiles over 5 material types. For each classifier, we used 20% for training, 5% for validation and 80% for testing. We found that the testing accuracy did not improve even if we used more than 20% data.

Spectrometer dataset

To learn discriminant spectral filters, we collected spectral profiles of several samples of everyday materials, namely, fabric, metal, paper, plant, plastic, skin, stone and wood. The data was collected under various lighting conditions such as sunlight, halogen light source and white LED light source. We divided the data capture into various session for a given lighting condition. We then collected spectra of materials, as well as spectral profile of a spectralon. This enabled us to estimate the reflectance of materials independent of illumination conditions.

Table 4.1 provides details of each material class. The two gray rows, metal and stone were not used in our experiments in the main paper. We found that metal and stone are best classified with a setup that works over visible to near IR ($400 - 900$nm) range, while our setup was only made for NIR ($600 - 900$nm). We plan to release the complete dataset, as well as our trained SVM and DNN classifiers for future research.

Support Vector Machine

We used the function `LinearSVC` from Scikit-Learn [76] for training binary classifiers. The only hyperparamter of tuning was penalty for the hyperplanes, C, which was tuned by performing a grid search over the log space from 10^3 to 10^6. Hyperparamter search was done through a 3-fold cross-validation.

Neural Networks

We used PyTorch [75] for training our neural network (DNN) classifiers. The architecture used for learning is shown in Fig. 4.6 and the details of each layer is provided in Table 4.2. Q is the number of filters and was varied from 1 to 20 to evaluate performance as a function of measurements. We trained the network with an initial learning rate of 10^{-3} and trained for a total of 500 epochs. The filters were initialized with a principal component analysis (PCA) decomposition of training data, which to smoother filters and higher accuracy. For each Q, we picked the model with best accuracy on validation dataset.

Data Class	Subclasses	Illumination conditions	#samples
Fabric	Cotton, Acrylic, Polyester, Wool	Halogen (VIS, NIR), LED (VIS)	2,258
Metal	Aluminum, Iron, Brass, Copper, Steel	Halogen (VIS, NIR), LED (VIS), Sunlight (VIS, NIR)	2945
Paper	A4 sheet, Novel pages, Book covers	Halogen (VIS, NIR), LED (VIS)	1024
Plant	Outdoor trees, Indoor plants, Succulents	Halogen (VIS, NIR), Sunlight (VIS, NIR)	2270
Plastic	HDPE, LDPE, Polyethene, Tire rubber	Halogen (VIS, NIR), LED (VIS), Sunlight (VIS, NIR)	1438
Skin	Palm, Forearm, Arm, Leg, Forehead	Halogen (VIS, NIR), LED (VIS), Sunlight (VIS, NIR)	400
Stone	Granite, Poured Concrete, Tar road	Halogen (VIS, NIR), Sunlight (VIS, NIR)	1495
Wood	Pine, Oak, Tree bark, Hardwood	Halogen (VIS, NIR), Sunlight (VIS, NIR)	2123

Table 4.1: **List of materials in dataset.** VIS implies the data has strong signal component in visible (400 – 900 nm) wavelengths, while NIR implies the data has strong signal component in 600 – 900nm regime.

Layer	Components
1 (**Filters**)	Linear (256xQ), ReLU, Dropout (0.1)
2	Linear (Qx256), ReLU, Dropout (0.1)
3	Linear (256x128), ReLU, Dropout (0.1)
4	Linear (128x64), ReLU, Dropout (0.1)
5	Linear (64x32), ReLU, Dropout (0.1)
6 (**Output**)	Linear (32x6), Softmax

Table 4.2: **Components of our DNN classifier.** All the layers are formed of fully connected layers with a ReLU and dropout added after each linear layer. Here, Q is the number of spectral filters and was variable in our experiments to compare performance. The output was a single linear layer. During training process, we used cross-entropy as loss function.

4.5.2 Optical setup

Hardware prototype. Figure 4.8 shows a picture of our lab prototype with names of major components. The last lens in the setup was replaced by a 50mm objective lens focused at infinity. This led to a better spatial resolution than an achromat.

Our SLM is a Holoeye LCoS SLM with a frame rate of 60 Hz that works as a secondary monitor. We used an NIR-sensitive sCMOS camera (Hamatasu ORCA Flash 4.0 LT). Inspired by previous work in material classification [25, 86], we designed our optical system to image from 600 – 900nm. Our setup is capable of coding spectrum at a resolution of 3.3nm, giving us 100 spectral bands. Finally, the SLM acts as a dynamic spectrally-selective camera and hence can be directly used for measuring the complete HSI. To do so, we display permuted Hadamard patterns on the SLM to capture a $256 \times 256 \times 256$ dimensional HSI. Figure 4.9 shows an example of captured HSI of an acrylic painting.

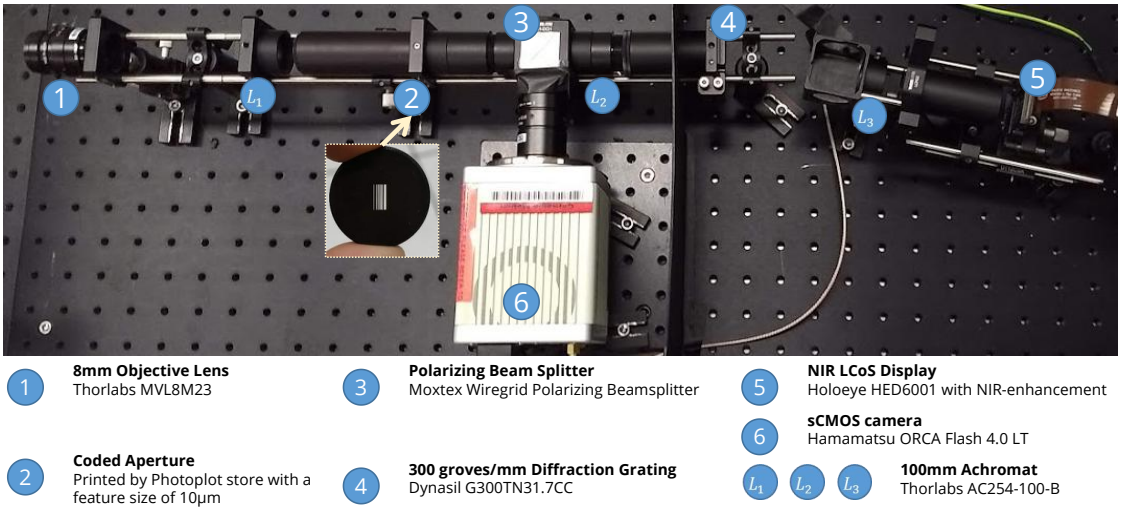
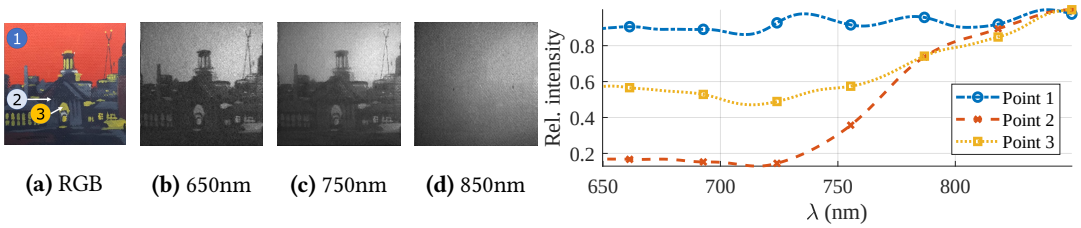


Figure 4.8: **Lab prototype.** A picture of the lab prototype along with major components marked with details. We skipped details about opto-mechanical components such as cage plates and posts to avoid clutter. The inset image shows the printed mask we used as coded aperture.

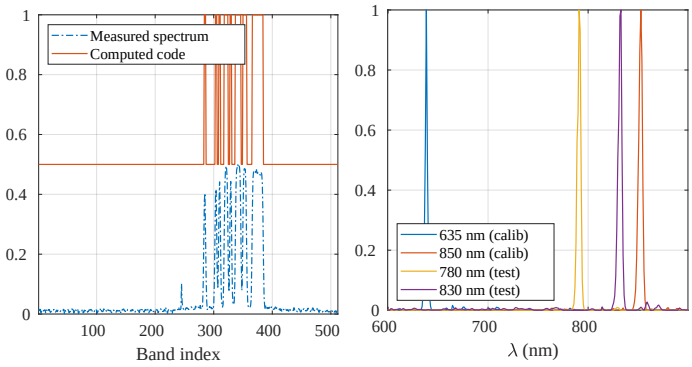


(a) RGB **(b)** 650nm **(c)** 750nm **(d)** 850nm

Figure 4.9: **Example HSI.** Our prototype is designed to capture images from 600nm to 900nm. (a) was captured using a cellphone while (b)-(d) are images captured by our setup. Bottom row shows spectral profiles at three marked points. Note how all the pigments disappear at $\lambda = 850$nm in (d).

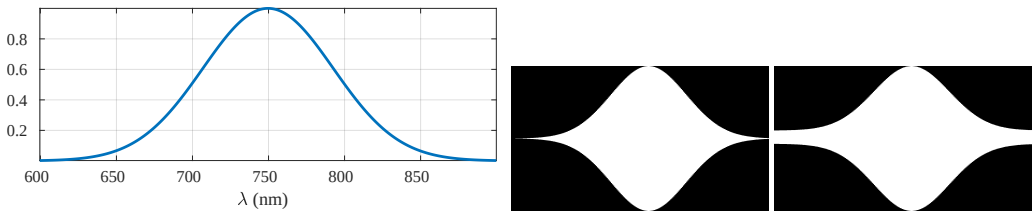
Calibration. As described in the previous sections, our setup required calibration of coded aperture, wavelengths and spatial PSF. We detail the calibration procedure here.

1. *Coded aperture calibration*: This is required to capture the code that blurs the spectrum. We measure the coded aperture by illuminating a spectrally flat object (such as spectralon) with a laser of known wavelength and scanning the complete HSI. We then average all spatial pixels to get the spectrum of the scene. Since a laser can be treated as a discrete delta, the measured spectrum will be the coded aperture. We threshold the measured spectrum appropriately to get the binary coded aperture, as shown in Fig. 4.10 (a).



(a) Code calibration **(b)** Wavelength calibration

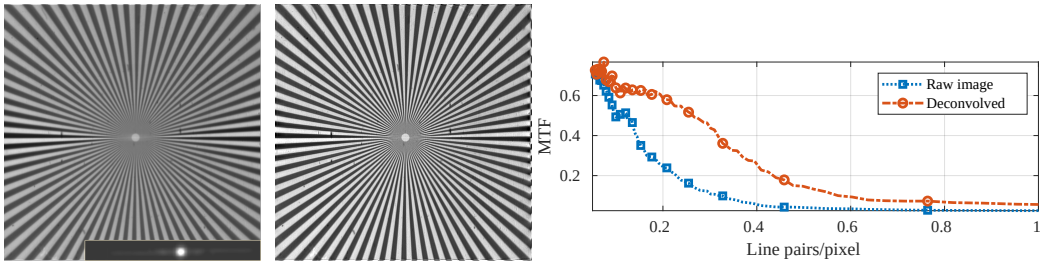
Figure 4.10: **Wavelength calibration.** We first estimate the blur due to coded aperture by capturing a scene illuminated by a narrowband light source (635nm laser), giving us the code in (a). We then calibrate the correspondence between band index and wavelengths by capturing two separate scenes illuminated by known laser light sources (635nm, 850nm). The results of the two calibration are show in (b), where we capture two more scenes with 780nm and 830nm laser.



(a) Target profile **(b)** Without correction **(c)** With correction

Figure 4.11: **Displaying desired spectral profile.** Given a target profile (a), we display a binary image on the SLM, as shown in (b), which ensures grayscale modulation despite wavelength-dependent gamma curve. However, since the SLM is $2f$ away from the camera sensor, there will be effects of diffraction. We counter this by adding a small DC offset, as shown in (c).

2. *Wavelength calibration*: To find the correspondence between band index (1 - 256) and the corresponding wavelengths, we capture two scenes, each one comprised of a spectrally flat object illuminated by a narrowband laser light source. The averaged spectrum of the HSI is a blurred version of the laser spectrum. By deconvolving with the previously estimated coded aperture, we get location of the laser in terms of band index. We use this information along with laser wavelength to calibrate the correspondence.



(a) Raw **(b)** Deconvolved **(c)** MTF

Figure 4.12: **Spatial deconvolution.** Due to design of an invertible spatial blur, the optical setup is capable of high resolution after deconvolution. (a) shows a raw image, with enlarged PSF in inset, (b) shows result of wiener deconvolution, and (c) shows a comparison of modulation transfer function (MTF). There is a marked increase in resolution both quantitatively and qualitatively.

3. *Spatial PSF*: To find the spatial blur kernel, we capture a single image of a 10μm pinhole. Since the PSF is well conditioned, deblurring the spatial images is well conditioned.

4. *Radiometric calibration of SLM*: The LCoS SLM in our optical setup is based on twisted-nematic design, and hence has different gamma curves for different wavelengths. Since the spectrum on the SLM is a blurred version of the true spectrum, we cannot perform a column-wise gamma correction. Instead, we use the SLM only as a binary modulator and achieve grayscale modulation by varying height of each column as shown in Fig 4.11 (b). This way, the SLM has a linear gamma curve for all wavelengths.

Figures of merit. Our setup is capable of achieving spectral resolution of up to 5nm over the wavelength range of 600 − 900nm, which is slightly worse than the designed resolution of 3.3 nm. (interested readers may refer to [87])). Due to invertible spatial blur, our setup is capable of high resolution after deconvolution. Figure 4.12 visualizes the captured image in (a) and deconvolved image in (b) of a sector star target. (c) shows plot of Modulation Transfer Function (MTF) as a function of line pairs per pixel. Image was deconvolved using simple Wiener deconvolution. The MTF30 after deconvolution was 0.45 linepairs/pixel.

Handling diffraction due to SLM. Since the SLM is placed $2f$ away from the image plane, any pattern displayed on SLM will lead to a diffraction blur. To counter this effect, we always display ones in the middle of the pattern to be displayed on the SLM. This reduces the effect of diffraction while adding a simple offset to the data, which can be removed by capturing image with only the central part open.

4.5.3 Handling illumination

The discussion so far has relied on classifying materials based on their *reflectance* spectrum only. In practice, we measure the reflectance multiplied by scene illumination as well as the camera's spectral response. Specifically, if $H(x, y, \lambda)$ is the HSI of the scene, then the *measured* HSI is $\widehat{H}(x, y, \lambda) = H(x, y, \lambda)c(\lambda)l(\lambda)$, where $c(\lambda)$ is the camera spectral response and $l(\lambda)$ is illuminant spectra. To account for the modified spectral measurements, we can follow one of the two approaches:

1. *Corrected spectral filters.* By displaying $\widehat{s}_k(\lambda) = s_k(\lambda)/(c(\lambda)l(\lambda))$, the measured spectral measurements are directly proportional to the reflectance spectra of the material.

2. *Training on modified data.* Instead of training on $H(x, y, \lambda)$ we can train on $\widehat{H}(x, y, \lambda)$, which then produces one model for each illuminant condition.

While both are equivalent, we found training on modified data to be more robust to illumination changes and hence trained different models for different experiments.

4.5.4 Dataset

To learn discriminant spectral filters, we collected spectra of several everyday object with a spectrometer[1]. For experiments in this chapter, we divided the materials into six classes, namely, fabric, paper, plants, plastic, human skin and wood – materials that are most likely to be found in everyday settings. We collected objects under varying illumination such as indoor illuminant, outdoor sunlight, LED light source and halogen light sources. All measurements were then divided with the corresponding illuminant spectrum to obtain reflectance spectra of objects. Image of some of the objects and spectral profiles are displayed in Fig. 4.2. We plan to release our dataset as well as the training code to the public. More details can be found in supplementary material.

4.5.5 Training classifiers

For binary classification, we used SVM to learn the spectral filters. We used Scikit-Learn [76] for this purpose. We trained DNNs for multi-class classification with the network architecture shown in Fig. 4.6 with loss function set to cross entropy. The number of spectral filters were varied from 1 to 20 to compare performance. We learned the network using the PyTorch framework [75] with learning rate set to 10^{-3} for a total of 500 epochs. We then extracted weights of first layer and used them as spectral

[1] OceanOptics Flame FX 400 – 1000nm spectrometer.

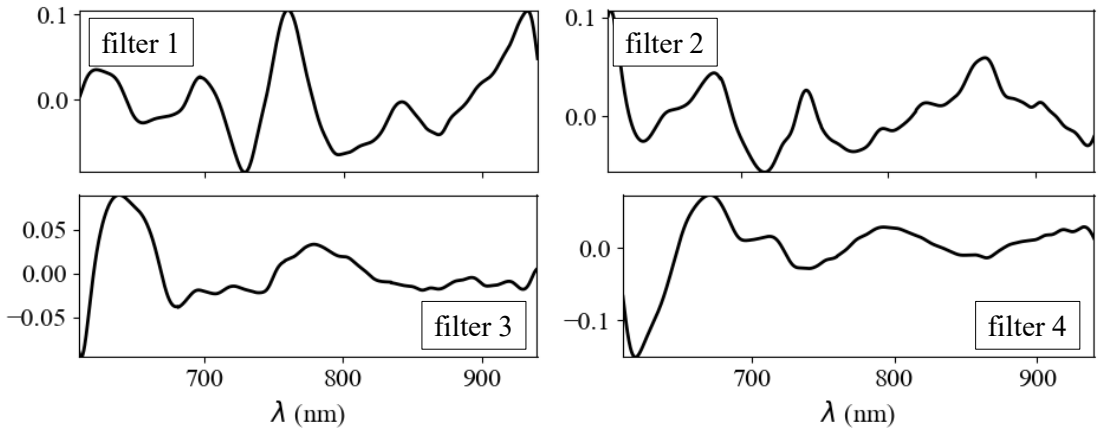


Figure 4.13: **Learned filters.** The filters shown here correspond to a DNN classifier with 4 spectral filters.

filters. The learned filters are shown in Fig. 4.13. The trained filters are then implemented on our lab prototype through a simple linear interpolation from spectrometer wavelengths to setup wavelengths. Further details about the learning process are included in supplementary.

4.5.6 Handling scale of features

A key requirement of any classifier is that the scale of features be same during training and testing. A common practice is to set the norm of feature at (x_0, y_0), $\|H(x_0, y_0, \lambda)\|$ to unity, or the maximum value to unity. In our case, this requires having knowledge of the complete spectral profile, which defeats the purpose of optical computing. instead, we normalize our measurements with sum of the spectrum, $\int_\lambda H(x_0, y_0, \lambda)$, which can be measured by displaying a spectral profile with all ones. The measured featured vectors are then,

$$I_{\text{sum}}(x_0, y_0) = \int_\lambda H(x_0, y_0, \lambda) d\lambda \tag{4.12}$$

$$\widetilde{I_k}(x_0, y_0) = \int_\lambda H(x_0, y_0, \lambda) s_k(\lambda) d\lambda \tag{4.13}$$

$$I_k(x_0, y_0) = \frac{\widetilde{I_k}(x_0, y_0)}{I_{\text{sum}}(x_0, y_0)} \tag{4.14}$$

Such an approach also makes the measurements invariant to shading of individual pixels [67–69]. We scale the spectra the same way even while training, which makes the scaling consistent. Hence any set of measurements with spectral profiles requires just one extra image.

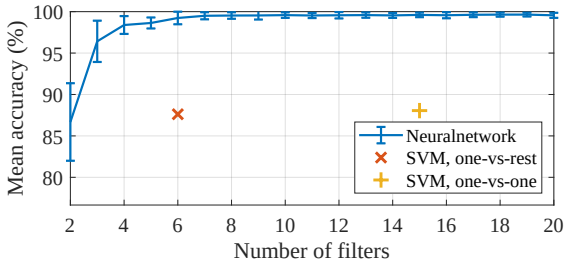


Figure 4.14: **Accuracy vs. number of filters.** We varied the number of filters and estimated mean accuracy on a held out test dataset. Based on the plot, we observe that DNNs outperform SVM, and that accuracy starts saturating beyond 4 filters.

4.5.7 Accuracy vs. number of filters

Classifiers such as SVM have a fixed number of filters, which is K for one-vs-rest and $K(K-1)/2$ for one-vs-one classifier. In contrast, neural networks can be designed with any number of spectral filters. To choose the appropriate filter count, we trained models with varying number of filters and evaluated average accuracy on a held-out test dataset. Fig. 4.14 shows an accuracy plot. We observe that DNNs achieve higher accuracy than SVM. Further, for our 6-class problem, we see that the accuracy saturate after four filters. We hence chose $Q = 4$ in all our experiments.

4.5.8 Accuracy vs. smoothness

Smoothness of spectral filters is controlled by regularization parameter in SVM and weight decay term in neural networks. It is desirable to have smooth spectral filters, as they lead to higher light throughput. In contrast, very smooth spectral filters may lead to lowered classification accuracy. To find the appropriate regularization term, we once again trained several models and chose the one that gave best accuracy on a held out dataset. Accuracy vs regularization for both SVM and DNNs is shown in Fig. 4.15.

4.5.9 Binary classification

The simplest task possible with our optical setup is a binary classification, where the label at each pixel belongs to one of the two possible classes. In such a situation, one may either use a linear SVM where the spectral filter is the learned supporting hyperplane, $\mathbf{w}$, or use a matched filter, where the spectral filter is difference of spectra of the two classes, $s_1(\lambda) - s_2(\lambda)$. Figure 4.16 shows classification of a real cactus surrounded by several plastic plants. The SVM score in (b) as well as the labels show that our setup is capable of resolving very thin structures such as the cactus thorns. Figure 4.17 shows classification

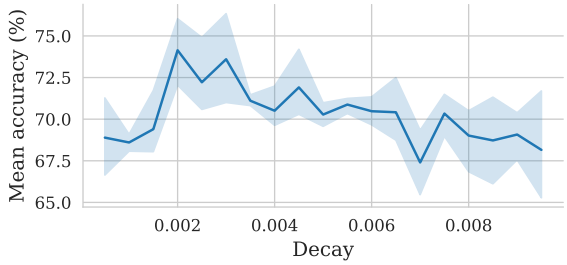
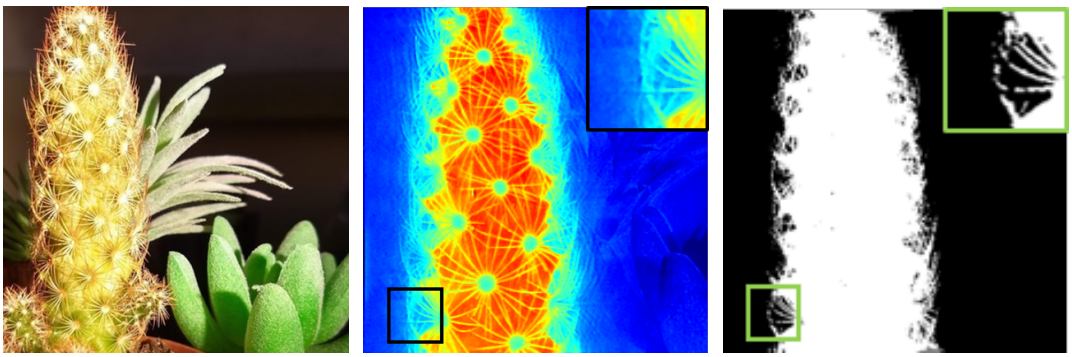


Figure 4.15: **Accuracy vs. smoothness.** We plotted accuracy as a function of decay constant, η, with lower η encouraging smoother filters. The accuracy is highest around $\eta = 2 \times 10^{-3}$, which was the value we used for all our experiments.



(a) RGB **(b)** SVM score **(c)** Label

Figure 4.16: **Per-pixel classification.** Due to per-pixel operation with high spatial resolution, our imager can clearly identify the micro-structures such as the cactus thorns by capturing only two images instead of the complete HSI.

results for real vs. plastic plants and real vs. fake wood with matched filtering and Fig. 4.18 shows classification results with SVM classifier.

Figure 4.19 evaluates the advantages of optical classification. (b) visualizes the SVM score at each pixel obtained by scanning the complete HSI and then computing the projection to the SVM hyperplane, which requires a total of 256 measurements. In contrast, optical projection, shown in (c) requires only two images. Bottom row shows the Receiver operating Characteristic (RoC) of classification for both cases. The SNR advantage is evident; the area under the curve for optical projection (0.7194) is higher than full measurement and then projection (0.7912).

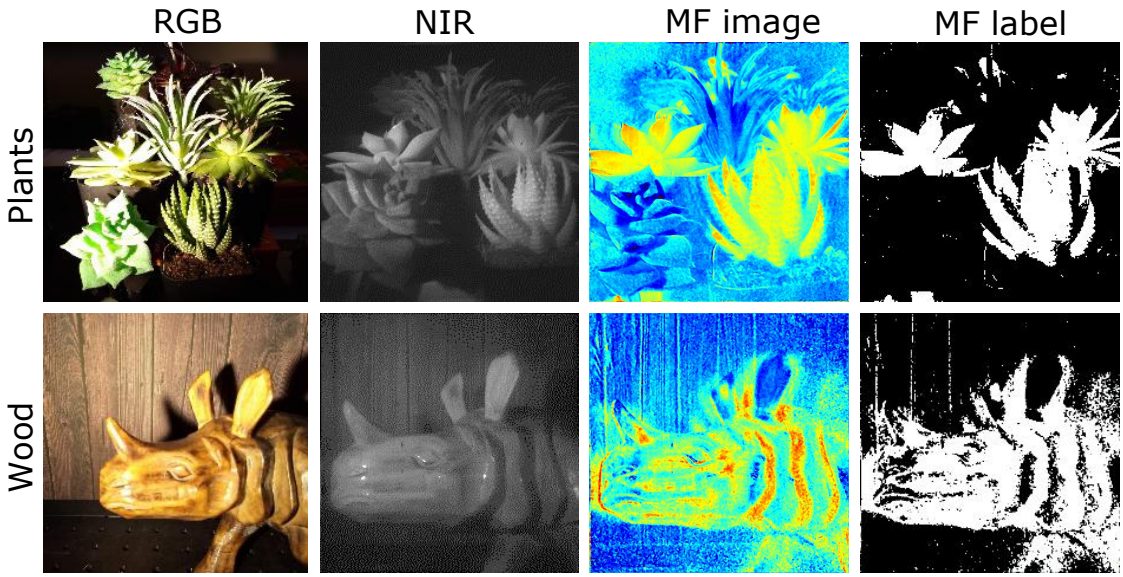


Figure 4.17: **Matched filtering.** If spectrum of the specific objects is known, one can implement a matched filter and perform per-pixel material classification.

4.5.10 Remote photoplethismography

By programming our optical setup to display profiles of oxygenated and deoxygenated blood, we can remotely perform pulse oximetry in a non-invasive way, which can be used for studying vasculature. Figure 4.21 shows an example data on index finger. We placed a light source behind the finger to capture data in transmissive mode, and collected 500 filtered images at 10 frames per second The obtained data was then temporally filtered to retain signal from $1 - 2$ Hz. The resultant data enhanced the digital arteries (dark line passing through center of the finger). The heart beat was estimated from this signal was 80.7 beats per minute, while an off-the-shelf pulse oximeter reported $79 - 84$ beats per minute.

4.5.11 Multi-class classification with DNNs

We first tested our prototype on an outdoor scene shown in Fig. 4.22 to verify that our setup was capable of accurate multi-class classification. We collected spectra with the prototype itself where we scanned full HSI of three classes, namely concrete, grass and sky, and then used the obtained spectral profiles to train a 3-class neural network. We then optically classified a scene with 6 filters. The resultant label map is shown in Fig. 4.22. Such a per-pixel label map can then be used for accurate semantic segmentation.

Next, we used the 6-class classifier that we trained earlier on spectrometer data (Fig. 4.13 and classified several indoor objects. Figure 4.23 compares the spectrally filtered output using full scan, as well as

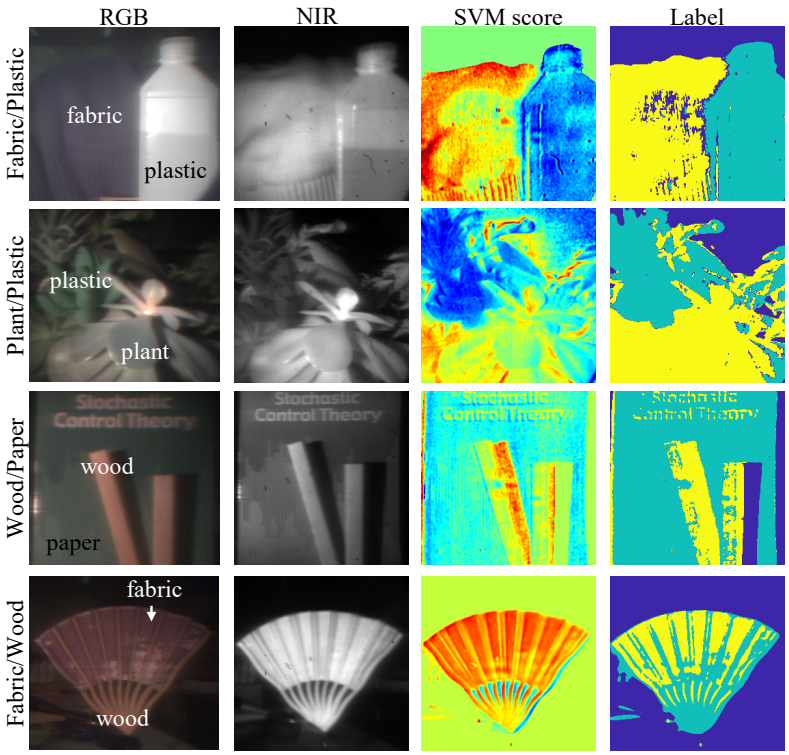
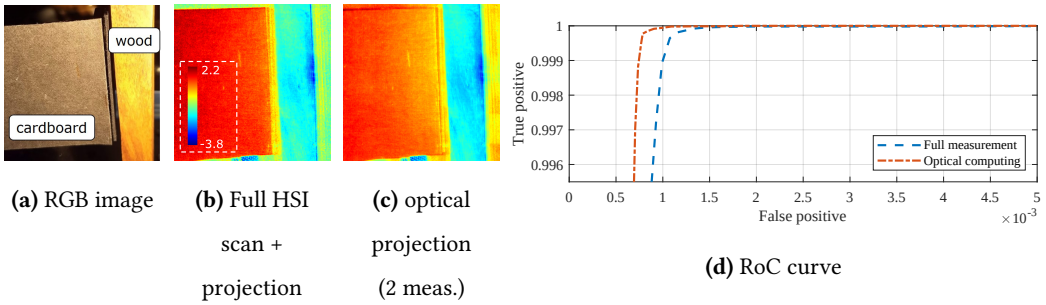


Figure 4.18: **SVM classification.** We learned SVM filters on spectrometer training data and then implemented them on our optical setup. Our setup is very versatile and can be used for classifying arbitrary material classes.



(a) RGB image

(b) Full HSI scan + projection (256 meas.)

(c) optical projection (2 meas.)

(d) RoC curve

Figure 4.19: **Advantage of optical computing.** We show an example of binary classification between cardboard and wood (a) using per-pixel SVM. Optical computing achieves higher accuracy with far fewer measurements.

optical projection. The two results look very similar; while optical computing requires only five images, full scanning required 256 images. Figure 4.24 shows several real world scenes classified with our optical classification strategy. In all cases, five images were captured, four with filters and one image with an

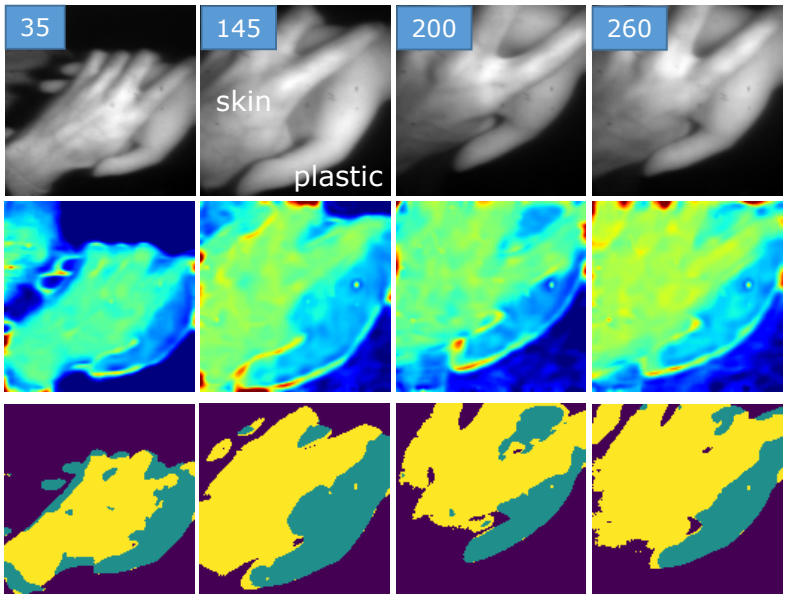


Figure 4.20: **Video rate classification.** Since our setup can rapidly change spectral filters, we can implement video-rate classification. In this example, we show some frames of a classifier that separates human skin from plastic (silicone in this example), which is often useful for biometrics.

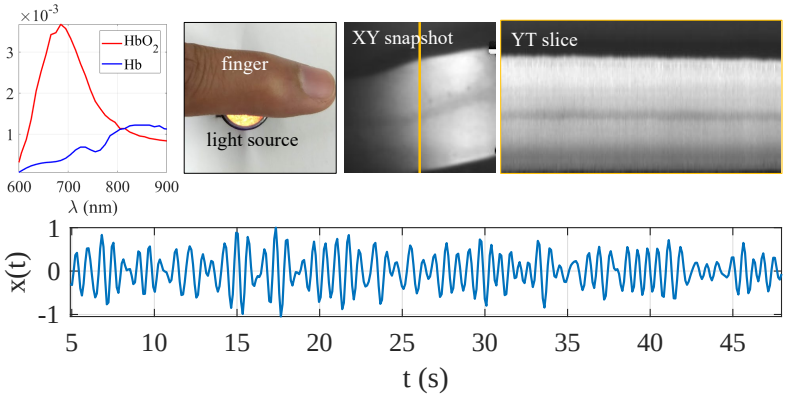


Figure 4.21: **Remote pulse oximetry.** Hemoglobin in blood varies between being oxygenated and deoxygenated, thereby having different spectral profiles (left plot). We program our camera with a difference of the two profiles, to capture oximetric signal in a remote, non-contact fashion. The plot in the second row shows the processed oximetry signal over the finger.

all-pass spectral filter.

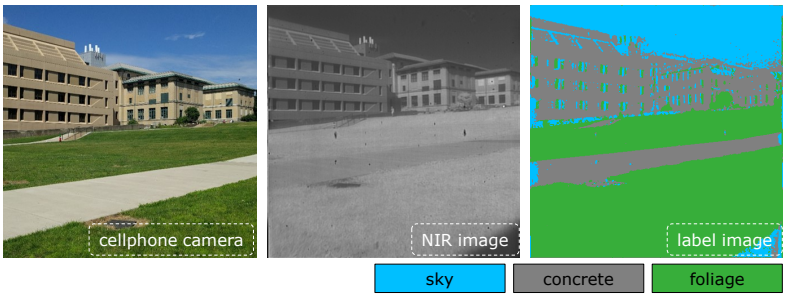
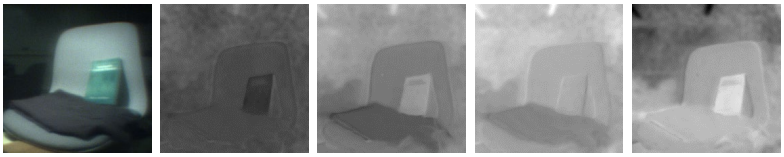
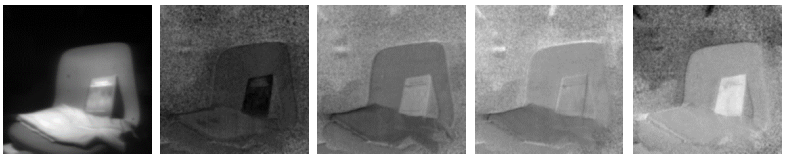


Figure 4.22: **Classification of outdoor materials with DNNs.** We trained a 3-class classifier with 6 filters to classify outdoor scene. Such a classification strategy can be used as an initial estimate for an accurate semantic segmentation.



(a) RGB (b) Filter 1, **full HSI** (c) Filter 2 (d) Filter 3 (e) Filter 4



(a) NIR (b) Filter 1, **optical** (c) Filter 2 (d) Filter 3 (e) Filter 4

Figure 4.23: n

Comparison of optical and full-scan spectral filtering. The proposed method is capable of directly measuring the spectral projections, which requires 5 images instead of 256 images (full HSI scanning).

4.5.12 Discussion

Across all the experiments, we note that the propose optical classification strategy is promising, particularly when operated in binary classification mode. We showed video-rate binary classification, as well as remote pulse oximetry. Importantly, we showed a *transferable* filter learning strategy where we learned spectral filters on spectrometer data and then implemented them on our setup. In case of multi-class classification, we note that the performance degrades compare to binary classification. This is to be expected due to two key reasons. One, any model mismatch arising due to the optical hardware propagates through the classification pipeline. Mismatches can cause severe error in estimates if spectral profiles are similar in shape. This can be corrected with careful and precise alignment of all optics.

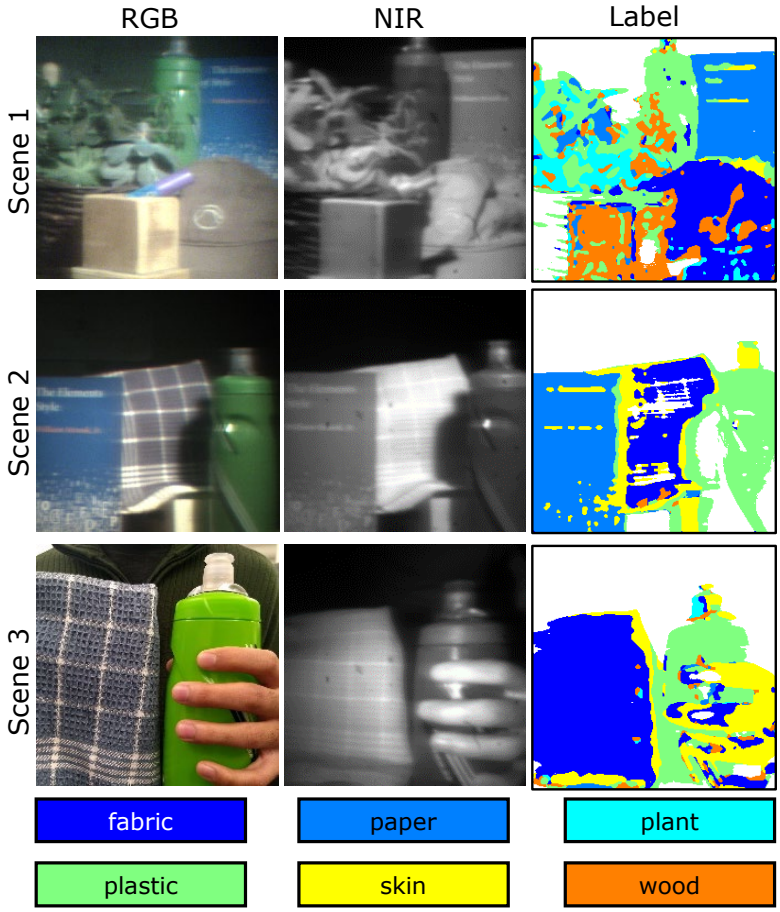


Figure 4.24: **Multiclass classification with DNNs.** We show diverse set of scenes that were classified with our optical classification strategy.

Second, higher order effects such as non-Lambertian surfaces, global illuminant component and scatter cause the measured spectrum to be different that the pure material spectra. Classification can be made robust if the training data is augmented with all possible spectral variations for each material. This can be achieved with a spectral-angular gantry, and will be pursued as a future direction.

4.5.13 Limitations

A key limitation of our setup is the assumption that the pixels come from a single material class. Some real world examples are made of a mixture of materials at each class, an example being land cover. In such a case, outputting just a class label may not suffice but relative probabilities of each class is desired. This can be achieved by modifying the classifiers to output a score for each material at each pixel instead

of most probable class.

4.6 Conclusion

We propose a per-pixel material classifier that relies on a high resolution programmable spectral filter. We achieve this by learning spectral filters that can achieve high classification accuracy and then measure images of the scene with the learned filters. Owing to a simple, per-pixel decoding strategy, we can achieve classification at video rates. We showed several compelling real world examples with emphasis on binary video-rate and multi-class classification. We also contributed a dataset of spectra of everyday materials which we believe will enable future research in spectrum-based material classification for computer vision.

5 Conclusions

This thesis focuses on efficient sensing and inference of hyperspectral images with novel optical systems. The core observation is that the true complexity of hyperspectral images lies in its spectral diversity, which is low for any given scene. The right approach is then to build tools that directly measure this low-dimensional approximation, without having to completely scan the scene's HSI. This requires us to first build an tool that is central to hyperspectral imaging – a spectrally programmable camera. Building such a camera was a challenging task, due to the space-spectrum uncertainty. We first see a concise theoretical expression for this uncertainty, and then demonstrate computational approaches to imaging beyond fundamental limits, along with engineered pupil functions.

The spectrally-programmable camera built in this thesis paves way to two important technologies which redefine the way we captures HSIs. The first one is an adaptive imager, that optically computes the low rank decomposition of the scene with as few as five images and five spectral measurements. The second one is an optical computing approach to classification, where fewer than five discriminant spectral filters are utilized to perform per-pixel classification.

This thesis emphasizes the importance of optical computing to tackle conventionally hard computer vision problems. Optical computing effectively transfers the computational burden to the sensing system, thereby resulting in significantly higher SNR, and remarkably few measurements. The idea of a computer in the loop for sensing goes beyond the applications in this thesis. We saw that learning schemes such as neural networks can be tightly coupled with optical systems by replacing the first layer with the camera output. Such end-to-end learning schemes will revolutionize how we sense very high dimensional signals. Imaging non-invasively under the skin, which faces the acute problem of heavy scattering is another hard task that is well tackled by optical computing. Some of the tools explored in this thesis, in particular spectral filtering, will prove invaluable to high-contrast imaging under such scattering conditions.

While we explored spectrum and space, which are crucial components of light and its interactions,

there are several other directions that require attention. For example, the work in this thesis relied on light being incoherent. However, several scientific applications such as microscopy and imaging via scattering medium rely on coherent imaging systems, and hence it is important to build optical computing and adaptive sensing tools for such settings. Similarly, the angular variations of light, captured as light fields, reveal important texture and micro-structure information about materials. As with spectrum, light fields are high dimensional and therefore would significantly benefit from the tools introduced in this thesis. To this end, sampling the complete *plenoptic function* [1] which involves sampling three spatial, two angular, one spectral and one temporal dimension is currently an unachievable task due to the enormous amount of data required for even modest number of samples along each axis. We believe this thesis makes contributions towards this grand goal.

We showed sampling and inference of HSIs at resolutions of several microns in spatial dimension and several nanometers in spectral dimension; however, the truest benefits are when the resolutions are at a much finer scale. Tasks such as Raman spectroscopy often require programming at much finer resolutions in order to accurately reveal the molecular structure. A future direction for this thesis would be to build imagers that work at such high resolutions. Another promising future direction is to utilize active illumination to optically compute the singular vectors of scene's HSI. A key benefit of such a system is that the spectral and spatial resolutions are decoupled, since illumination is separate from the spatial imager. High resolution hyperspectral imaging with active illumination systems will be indispensable tools in microscopy where fast and accurate estimation of properties of the specimen is crucial.

The past decade focused on building imagers that capture compressed measurements, and then *separately* use computational tools to recover the signal. I believe this thesis sets tone for a new set of imagers that utilize signal models to design sensing technologies, in a closed loop fashion. To this end, I envision that the ideas presented in this paper is a step towards bring the hyperspectral camera to mainstream computer vision applications, reducing the gap between what we *can* measure and what we *should* measure.

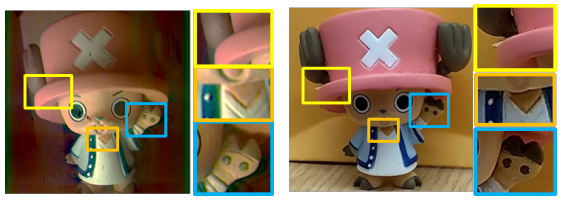
A Supplementary for KRISM

A.1 Real experiments

We provide visualizations for some of the real experiments presented in the previous sections. Specifically, we compare the captured singular vectors for two scenes with spectrally Hadamard multiplexed measurements. We also show spectral band images for Macbeth chart and crayons chart, showing the intensity variation of various colors.

A.1.1 Visualizing spatial images

Figure A.3 shows images across various wavelengths for the “color checker" scene and “crayons" scene. In particular, The images show the variation of intensity of each color swatch/crayon across wavelengths, with blur objects being brighter initially, green objects in the middle and red objects finally. Figure A.1 shows RGB image captured by our lab prototype as well as a cellphone camera for the Chopper scene. The insets shows textured areas which show high spatial resolution of our prototype.



(a) KRISM **(b)** Cellphone

Figure A.1: **Comparison of RGB images with our lab prototype and cellphone**. Note that the KRISM image is not white balanced whereas the cellphone image is white balanced by default. The zoomed in patches show edges in the toy, establishing high spatial resolution of our lab prototype.

A.1.2 Comparison of singular values and singular vectors

The ability of KRISM to accurately compute singular vectors has been presented in the previous sections. Here, we present two more experimental measurements to show how KRISM is applicable across various settings. Comparison is done against spectrally Hadamard multiplexed data, and then computing singular vectors on computer. We evaluate three metrics, namely, SNR between singular values, SAM between spectral singular vectors and SAM between spatial singular vectors. “Color checker" experiment (first row in Figure A.2) was captured by placing the Macbeth chart in front of the camera, and illuminating with a tungsten-halogen light source. The PSNR between singular values was 45.8dB, average SAM between spectral singular vectors 10° and that between spatial singular vectors was 10°. “Chopper" experiment (second row in Figure A.2) was captured by placing the Chopper toy in front of the camera, and illuminating it with CFL, a peaky illuminant. The PSNR between singular values was 39.3dB, average SAM between spectral singular vectors 10° and that between spatial singular vectors was 10°. Finally, the last row shows a comparison between singular values from Hadamard multiplexing and singular values from KRISM for some scenes presented in the previous sections. Across the board, KRISM computes the low-rank approximation with very high accuracy, as is evident from the experiments.

A.2 Synthetic experiments

We showed some simulation results in the previous sections. We show several more examples here, with emphasis on diversity of datasets. We tested KRISM via simulations on four different datasets and compared it against competing techniques for hyperspectral imaging.

Datasets. We used the hyperspectral data set by Arad and Ben-Shahar [2] (ICVL dataset), which consists of several high spatial and spectral resolution hyperspectral images covering 519 bands in visible and near IR wavelengths. We downsampled the HSI to $256 \times 256 \times 260$ to keep computation with CASSI-type simulations tractable. We also used datasets from Choi *et al.* [18], Yasuma *et al.* [102] and Chakrabarti and Zickler [15] with 31 spectral bands to compare with learning-based techniques. Finally, we present one example from NASA AVIRIS to compare KRISM against Row/Column CS proposed in [27].

Competing methods. We compared KRISM against four competing CS hyperspectral imaging techniques. All methods were simulated with 60dB readout and photon noise and 12-bit quantization.

Specifics of each simulation model are given below:

1. *KRISM*: We performed a rank-4 approximation of the HSI with 6 spatial and 6 spectral measurements. Diffraction blur due to coded aperture was introduced both in spectral and spatial profiles. Deconvolution was then done using Wiener deconvolution in both spectral and spatial domains.

2. *Fazel et al.* [27]: As with KRISM, we performed a rank-4 approximation of the HSI by computing random Gaussian projections with 6 spatial and 6 spectral measurements. Diffraction blur due to coded aperture was introduced as well.

3. *Lin et al.* [57]: We recovered HSI from a single snapshot image using technique in Lin *et al.* [57].

4. *Choi et al.* [18]: We recovered HSI from a single snapshot image using technique in [18].

5. *Kittle et al.* [49]: We used the multi-frame CASSI architecture for obtaining coded images, and recovered the HSI with sparsity prior in wavelet domain. We reduced the number of spectral bands for ICVL dataset to 31 to keep computations tractable.

6. *Sun and Kelly* [93]: We obtained spatially-multiplexed spectral measurements with random permuted Hadamard matrix and recovered the HSI with sparsity in wavelet domain.

We define reconstruction SNR as $\text{rsnr} = 20\log_{10}\left(\frac{\|\mathbf{x}\|_F}{\|\mathbf{x}-\hat{\mathbf{x}}\|_F}\right)$, where $\|\cdot\|_F$ is the Frobenius norm and $\hat{\mathbf{x}}$ is the recovered version of $\mathbf{x}$.

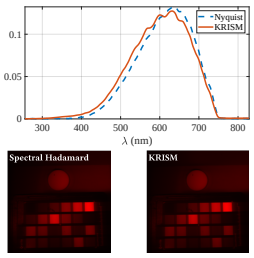
A.2.1 Performance with high spectral resolution

The true potential of KRISM can be exploited when there are a large number of spectral bands, such as the ones in the dataset by Arad and Ben-Shahar [2]. We keep compression low for competing methods as the accuracy scaled poorly with higher compressions (see Figure A.7) Results on some representative examples have been show in Figure A.4. Qualitatively, the reconstructed spatial images as well as the spectral signatures are very close to ground truth. Figure A.5 shows a comparison of reconstruction SNR as a function of compression ratios. As is evident, KRISM works significantly better than other methods despite very high compression ratios. The closes competitor to KRISM is the Row/Column CS approach by [27]. We show comparison between KRISM and Row/Column CS on one example from NASA's AVIRIS dataset consisting of of 224 spectral bands between 400-2400nm, and on SpecTIR [92] dataset consisting of 178 spectral bands, making it a good example to test our method. Results are shown in Figure A.6. For the same compression ration, KRISM offers a 10dB higher accuracy, and is qualitatively more accurate in both spatial images and spectral profiles.

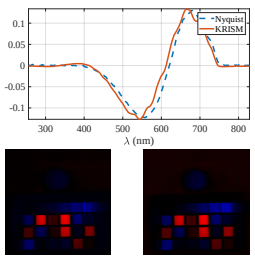
A.2.2 Performance with low spectral resolution

Most of the visible HSI datasets contains 31-33 spectral bands between 400 - 700nm. In this regime, learning-based snapshot techniques such as Choi *et al.* [18] and [57] have better performance. We used the dataset by Chakrabarti and Zickler [15], Choi *et al.* [18] and Yasuma *et al.* [102] for simulations with 31 spectral bands. Spatial resolution has been specified for individual images in Figure A.8. We compare KRISM with varying number of measurements against snapshot techniques [18, 57] in Figure A.9. We observe that in the setting closest to snapshot mode, Choi *et al.* [18] and Lin *et al.* [57] do outperform KRISM; this is to be expected since after a single iteration, KRISM provides only a rank-1 approximation. As the number of KRISM iterations are increased (which allows approximations of higher ranks), KRISM performance improves. We note that simulations for [57] were performed with downsampled dataset and only on a select set of scenes as the recovery required several days for each scene even with parallelization. Figure A.10 shows recovery SNR as a function of compression for multi-frame techniques. As was discussed earlier, KRISM is particularly effective for high resolution imaging. However, even with small number of bands, performance is superior in terms of spatial and spectral resolutions. It is worth noting that learning-based snapshot technique by [18] outperforms [49] with fewer measurements. This is expected, as it exploits the smooth nature of underlying spectra.

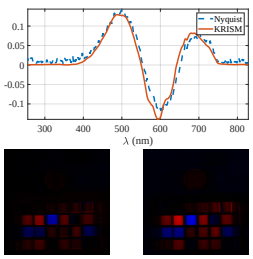
(a) “Color checker" scene



(b) First singular vector

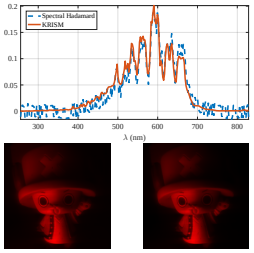


(c) Second singular vector

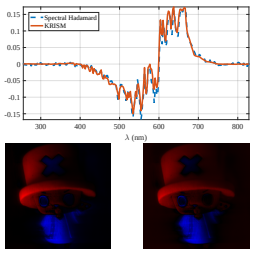


(d) Third singular vector

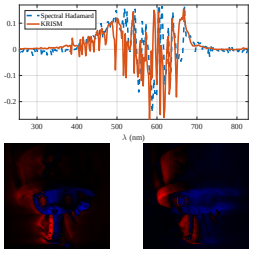
(e) “Chopper" scene



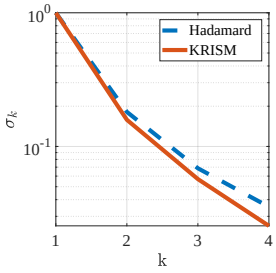
(f) First singular vector



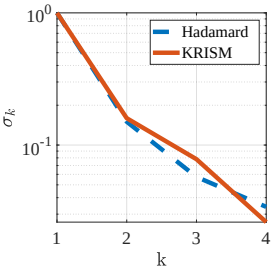
(g) Second singular vector



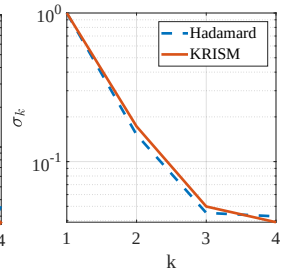
(h) Third singular vector



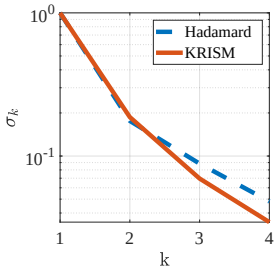
(i) Dice (PSNR = 37.2dB)



(j) Color checker (PSNR = 45.8dB)



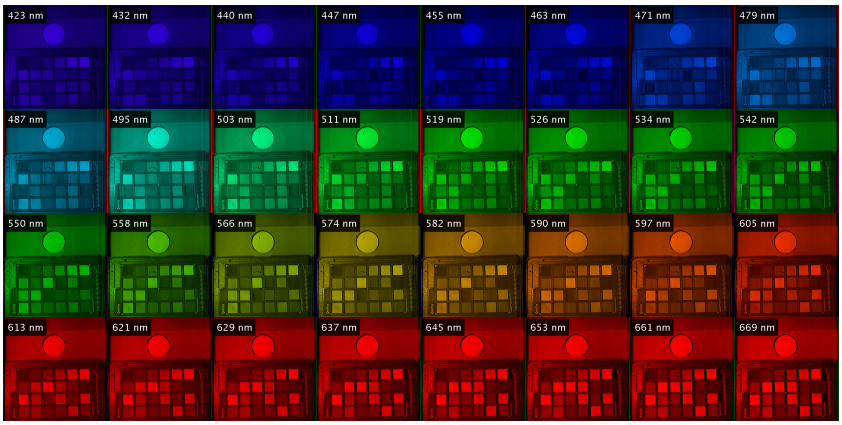
(k) Chopper (PSNR = 39.3dB)



(l) Crayons (PSNR = 37.8dB)

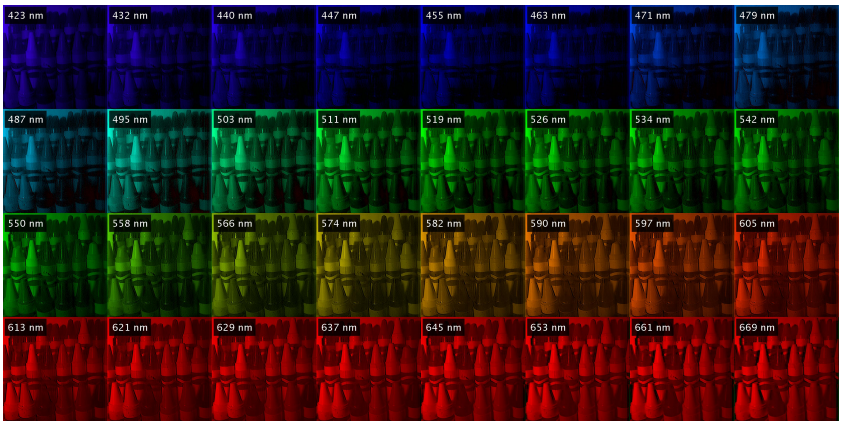
Figure A.2: **Comparison of singular values and singular vectors captured via spectrally Hadamard-multiplexed sensing and KRISM**. The left image singular vector is from Hadamard multiplexed data and the right one is from KRISM. Blue represents negative values and red represents positive values. KRISM method required capturing a total of 6 spectral and 6 spatial measurements to construct 4 singular vectors. While the Nyquist sampling method took a total of 59 minutes, KRISM took under 5 minutes. Top row shows singular vectors for “Color checker" scene, the middle row shows singular vectors for “Chopper" scene, and the last row compares singular values computed by Hadamard multiplexing and KRISM. Overall, our optical setup captures a low-rank approximation of the HSI with high accuracy.

(a) “Color checker" scene



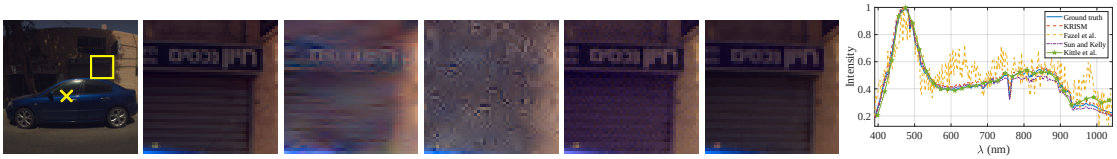
(b) Spectral band images

(c) “Crayons" scene



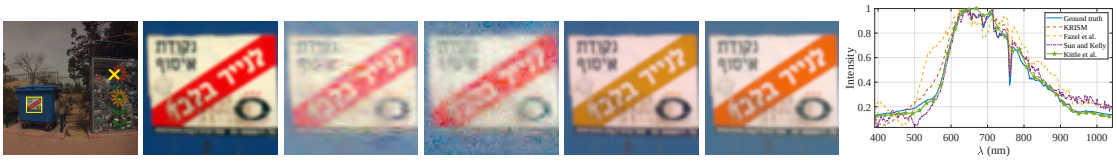
(d) Spectral band images

Figure A.3: **Crayons and Macbeth scene images across different wavelengths**. Both scenes were captured by illuminating the subjects with a tungsten-halogen bulb and then obtaining a rank-4 approximation with KRISM. The intensity of the crayons across wavelengths correctly reflects the individual color.



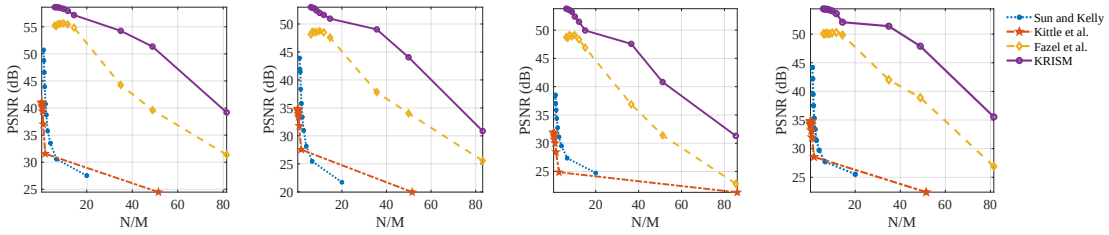
(a) Arad and Ben-Shahar ($256 \times 256 \times 260$) **(b)** Ground truth **(c)** Kittle et al. PSNR: 38.6dB, $N/M : 5$ **(d)** Sun and Kelly PSNR: 41.1dB, $N/M : 5$ **(e)** Fazel et al. PSNR: 40.3dB, $N/M : 43$ **(f)** KRISM PSNR: 49.1dB, $N/M : 43$ **(g)** Spectrum at marked point

(a) Arad and Ben-Shahar ($256 \times 256 \times 260$) **(b)** Ground truth **(c)** Kittle et al. PSNR: 33.3dB, $N/M : 5$ **(d)** Sun and Kelly PSNR: 34.0dB, $N/M : 5$ **(e)** Fazel et al. PSNR: 35.4dB, $N/M : 43$ **(f)** KRISM PSNR: 39.9dB, $N/M : 43$ **(g)** Spectrum at marked point



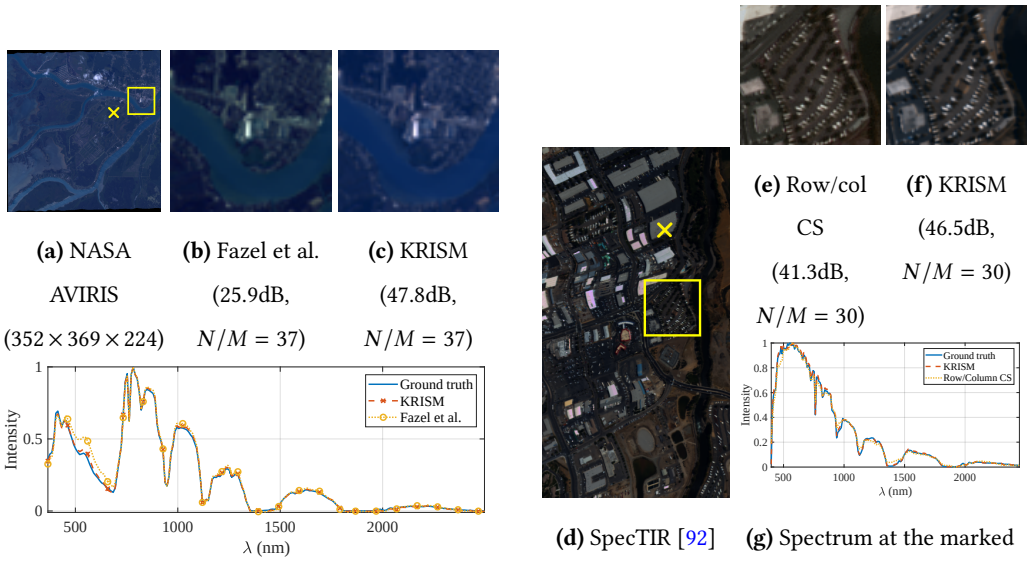
(a) Arad and Ben-Shahar ($512 \times 512 \times 260$) **(b)** Ground truth **(c)** Kittle et al. PSNR: 27.1dB, $N/M : 5$ **(d)** Sun and Kelly PSNR: 30.1dB, $N/M : 5$ **(e)** Fazel et al. PSNR: 36.8dB, $N/M : 43$ **(f)** KRISM PSNR: 44.1dB, $N/M : 43$ **(g)** Spectrum at marked point

Figure A.4: **Simulations on high resolution datasets.** Visualization of results for the high resolution dataset by [2]. "CASSI" represents Single Disperser CASSI, recovered using spectral prior [18]. Kittle et al. uses multiple spatio-spectral images [49], and was reconstructed with sparsity in wavelet domain. Sun and Kelly represents spatially-multiplexed measurements [93], and was reconstructed with sparsity in wavelet domain. Row/Col CS represents random row and column projections [27], and KRISM is the proposed method. Simulations were performed with 60dB readout, and photon noise. Row/Col CS and KRISM were simulated with spatio-spectral diffraction blur. We show zoomed in image patches for each method and spectrum at pixel marked by a cross. Across the board, KRISM outperforms all methods, both qualitatively and quantitatively.



(a) Car scene **(b)** Checker scene **(c)** Dustbin scene **(d)** Bulb scene

Figure A.5: **Accuracy vs. compression ratio**. Comparison of reconstruction SNR vs compression ratio for various methods on [2] dataset. Simulations were done as described in Figure A.4. KRISM outperforms any other method by a larger margin in both approximation accuracy, as well as compression ratio.



(a) NASA AVIRIS ($352 \times 369 \times 224$) **(b)** Fazel et al. (25.9dB, $N/M = 37$) **(c)** KRISM (47.8dB, $N/M = 37$)

(e) Row/col CS (41.3dB, $N/M = 30$) **(f)** KRISM (46.5dB, $N/M = 30$)

(d) SpecTIR [92] ($320 \times 600 \times 178$) **(g)** Spectrum at the marked location.

Figure A.6: **KRISM vs Row/Col CS**. Comparision of Row/Col CS vs KRISM for large number of spectral bands with high spatial and spectral resolution, making a good candidate for KRISM. Simulations were done with 60dB readout noise, photon noise, and diffraction blur on spatial images and spectra. We show zoomed in image patches for each method and spectrum at pixel marked by a cross. For the same compression ratio, KRISM outperforms Row/Col CS by 10dB.

(a) Ground truth **(b)** KRISM 47.7dB **(c)** Sun et al. 32.5dB **(d)** Kittle et al. 30.2dB

Figure A.7: **Performance of multi-frame methods at high compression**. We show a simulated example of recovery with $N/M = 43$ with (a) KRISM, (b) Sun and Kelly [93] and (c) Kittle *et al*. [49]. Existing multi-frame techniques do not work well under high compression ratio. While KRISM recovers spatial images accurately, [49, 93] lead to severe loss in resolution.

(a) KAIST dataset (512×384×31)

(b) Choi et al. PSNR: 35.0dB, N/M : 29

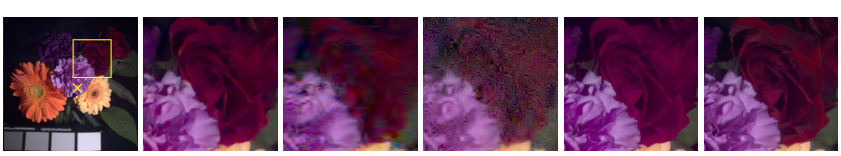
(c) Kittle et al. PSNR: 32.4dB, N/M : 5

(d) Sun and Kelly PSNR: 31.2dB, N/M : 5

(e) Fazel et al. PSNR: 30.6dB, N/M : 5

(f) KRISM PSNR: 37.5dB, N/M : 5

(g) Spectrum at marked point



(a) Yasuma et al. (512×512×31)

(b) Choi et al. PSNR: 37.7dB, N/M : 29

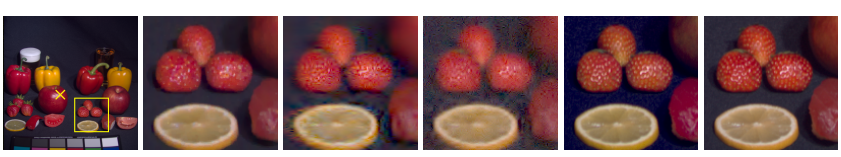
(c) Kittle et al. PSNR: 35.2dB, N/M : 5

(d) Sun and Kelly PSNR: 31.6dB, N/M : 5

(e) Fazel et al. PSNR: 40.6dB, N/M : 5

(f) KRISM PSNR: 48.2dB, N/M : 5

(g) Spectrum at marked point



(a) Yasuma et al. (512×512×31)

(b) Choi et al. PSNR: 37.6dB, N/M : 29

(c) Kittle et al. PSNR: 36.4dB, N/M : 5

(d) Sun and Kelly PSNR: 32.2dB, N/M : 5

(e) Fazel et al. PSNR: 32.8dB, N/M : 5

(f) KRISM PSNR: 41.2dB, N/M : 5

(g) Spectrum at marked point

(a) Chakrabarti et al. (696×520×31)

(b) Choi et al. PSNR: 42.0dB, N/M : 29

(c) Kittle et al. PSNR: 38.9dB, N/M : 5

(d) Sun and Kelly PSNR: 44.5dB, N/M : 5

(e) Fazel et al. PSNR: 45.6dB, N/M : 5

(f) KRISM PSNR: 50.7dB, N/M : 5

(g) Spectrum at marked point

(a) Chakrabarti et al. (696×520×31)

(b) Choi et al. PSNR: 41.3dB, N/M : 29

(c) Kittle et al. PSNR: 36.0dB, N/M : 5

(d) Sun and Kelly PSNR: 46.9dB, N/M : 5

(e) Fazel et al. PSNR: 47.9dB, N/M : 5

(f) KRISM PSNR: 50.2dB, N/M : 5

(g) Spectrum at marked point

Figure A.8: **Comparison of reconstructed images low spectral resolution**. All experiments were performed with 60dB readout noise and poisson noise. We show zoomed in image patches for each method and spectrum at pixel marked by a cross. For lower spectral resolution, KRISM offers limited benefits in compression ratios but is superior in terms of spatial and spectral reconstructions and overall accuracy.

(a) Lin et al. (512 × 384 × 31)

(b) Ground truth

(c) Lin et. al N/M : 29, PSNR: 26.6dB

(d) Choi et al. N/M : 29, PSNR: 33.3dB

(e) KRISM K : 1 N/M : 10, PSNR: 28.3dB

(f) KRISM K : 2 N/M : 8, PSNR: 37.7dB

(g) KRISM K : 3 N/M : 6, PSNR: 41.1dB

(h) KRISM K : 4 N/M : 5, PSNR: 44.2dB

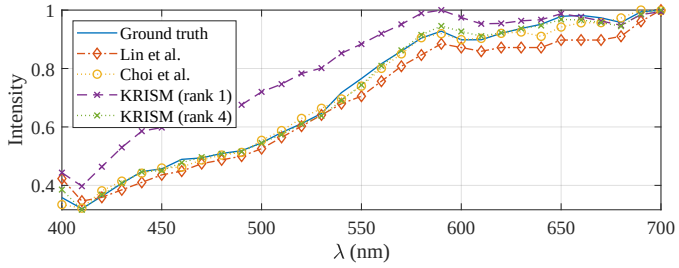
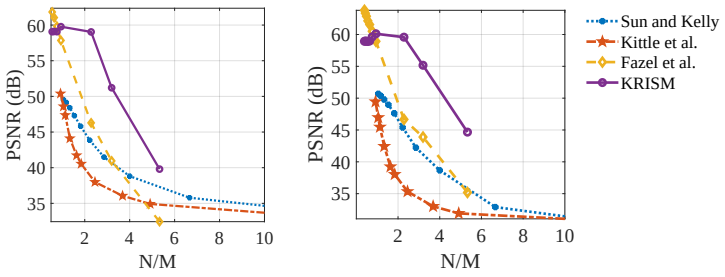


Figure A.9: **Evaluation on Door dataset from [15]**. We compared KRISM against methods from [57] and [18] on low-resolution spectra. We show zoomed in image patches for each method and spectrum at pixel marked by a cross. At settings close to snapshot sensing ($K = 1$), data-driven techniques perform better; with more iterations, KRISM achieves higher quality in spatial and spectral resolution.



(a) Outdoor scene **(b)** Books scene

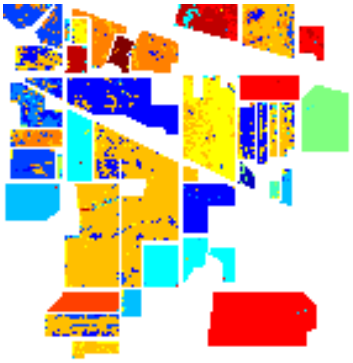
Figure A.10: **Accuracy vs. compression for low-resolution datasets**. Comparison of reconstruction SNR vs compression ratio for various methods on [15] dataset. Simulations were done as described in Figure A.8. Despite lower compression ratios, KRISM promises greater overall performance.

B Supplementary for Programmable Spectrometry

B.1 Simulations

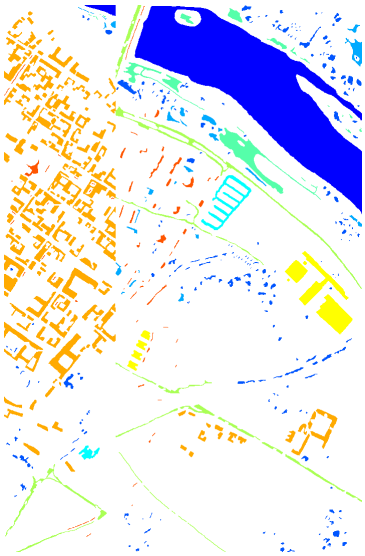
We showed simulations on the “Indian Pine" dataset from NASA where the average accuracy was compared with state-of-art techniques in hyperspectral classification. Our simulations showed that using spectral features alone reduced accuracy by a small amount but required far more measurements than the number of bands in the HSI. We show simulations on some more datasets in comparison to state-of-art results along with visualization of labels in Fig. B.1. We used the same models for SVM and DNN as our real experiments. To keep comparison fair between SVM and DNN, we set the number of spectral filters to number of object classes. Overall, DNN outperforms SVM, often by a large margin. More importantly, DNN is capable of achieving very high accuracy even when the first layer of the network is constrained to be linear, and fully connected. This validates our hardware for classification tasks.

(a) Indian Pine

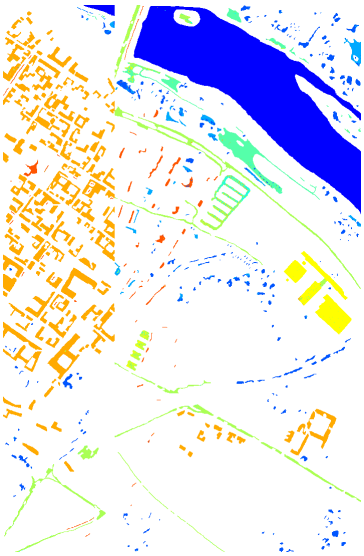


(b) SVM (83.1%)

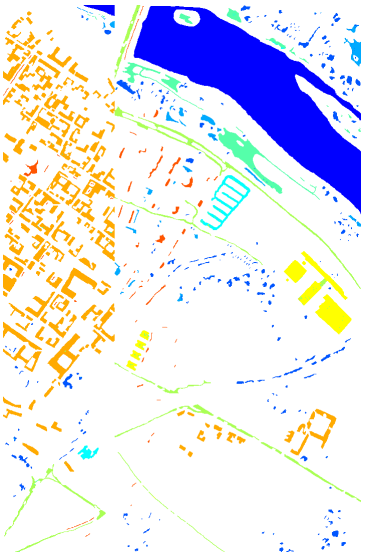
(c) DNN (93.0%)



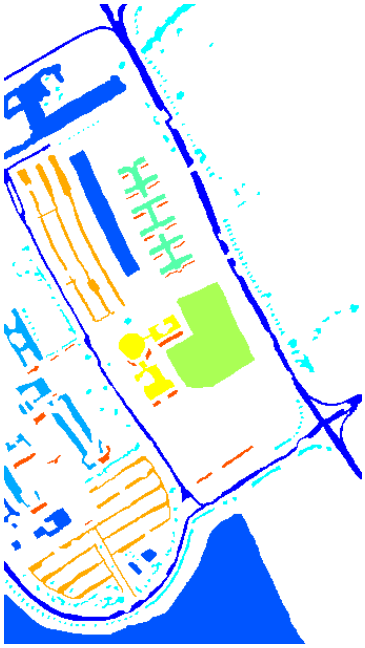
(d) Pavia center



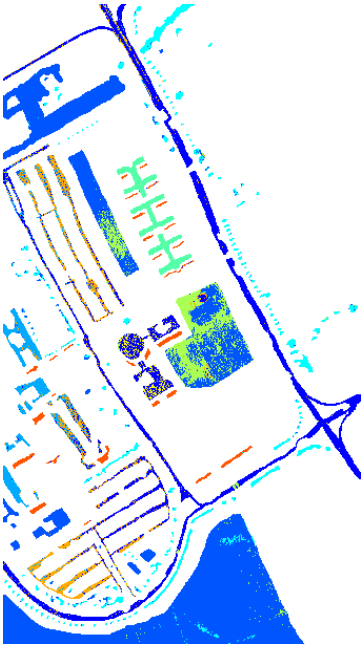
(e) SVM (96.5%)



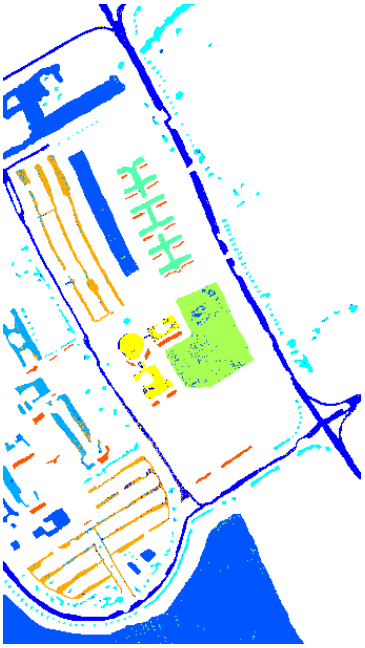
(f) DNN (99.4%)



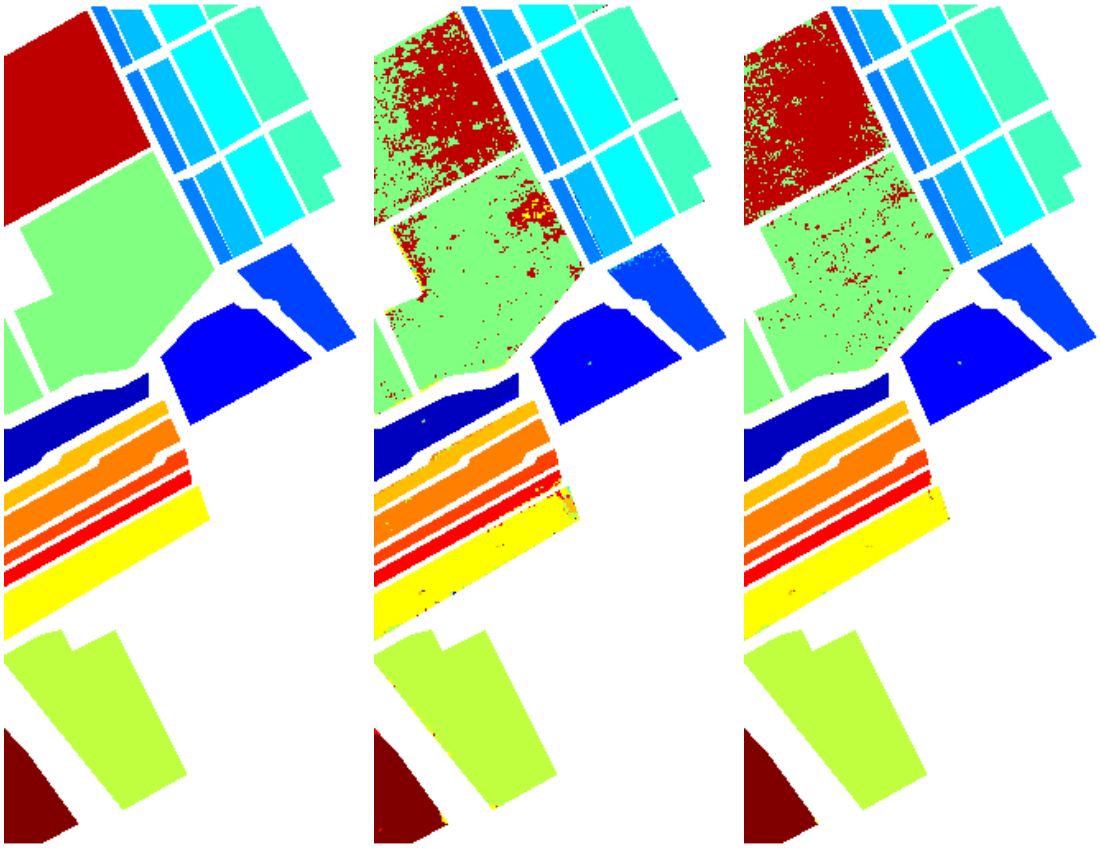
(g) Pavia center



(h) SVM (82.2%)



(i) DNN (96.7%)



(j) Salinas **(k)** SVM (91.6%) **(l)** DNN (97.3%)

Figure B.1: **Performance comparison.** Each dataset was trained with same number of spectral filters as number of object classes. Across the board, DNN has higher accuracy than SVM and can achieve near-perfect classification.

Bibliography

[1] Edward H Adelson, James R Bergen, *et al.* 1991. *The plenoptic function and the elements of early vision.* Vision and Modeling Group, Media Laboratory, Massachusetts Institute of Technology. 96

[2] Boaz Arad and Ohad Ben-Shahar. 2016. Sparse Recovery of Hyperspectral Signal from Natural RGB Images. In *European Conf. Computer Vision.* 98, 99, 103, 104

[3] Gonzalo R Arce, David J Brady, Lawrence Carin, Henry Arguello, and David S Kittle. 2014. Compressive coded aperture spectral imaging: An introduction. *IEEE Signal Processing Magazine* 31, 1 (2014), 105–115. 32

[4] Henry Arguello and Gonzalo R Arce. 2013. Rank minimization code aperture design for spectrally selective compressive imaging. *IEEE Trans. Image Processing* 22, 3 (2013), 941–954. 36

[5] Ery Arias-Castro and Yonina C Eldar. 2011. Noise folding in compressed sensing. *IEEE Signal Processing Letters* 18, 8 (2011), 478–481. 4

[6] Ravindra A Athale and William C Collins. 1982. Optical matrix–matrix multiplier based on outer product decomposition. *Appl. optics* 21, 12 (1982), 2089–2090. 36

[7] Yitzhak August, Chaim Vachman, Yair Rivenson, and Adrian Stern. 2013. Compressive hyperspectral imaging by random separable projections in both the spatial and the spectral domains. *Appl. Optics* 52, 10 (2013), D46–D54. 4, 7, 15

[8] Seung-Hwan Baek, Incheol Kim, Diego Gutierrez, and Min H Kim. 2017. Compact single-shot hyperspectral imaging using a prism. *ACM Trans. Graphics* 36, 6 (2017), 217. 35

[9] Richard G Baraniuk. 2007. Compressive sensing. *IEEE Signal Processing Magazine* 24, 4 (2007), 118–121. 4, 35

[10] Richard G Baraniuk, Volkan Cevher, Marco F Duarte, and Chinmay Hegde. 2010. Model-based compressive sensing. *IEEE Trans. Info. Theory* 56, 4 (2010), 1982–2001. 5

[11] Marion F. Baumgardner, Larry L. Biehl, and David A. Landgrebe. 2015. 220 Band AVIRIS Hyperspectral Image Data Set: June 12, 1992 Indian Pine Test Site 3. https://doi.org/doi:/10.4231/R7RX991C 2

[12] José M Bioucas-Dias and Mário AT Figueiredo. 2007. A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration. *IEEE Trans. Image processing* 16, 12 (2007), 2992–3004. 62

[13] Xun Cao, Tao Yue, Xing Lin, Stephen Lin, Xin Yuan, Qionghai Dai, Lawrence Carin, and David J Brady. 2016. Computational snapshot multispectral cameras: toward dynamic capture of the spectral world. *IEEE Signal Processing Magazine* 33, 5 (2016), 95–108. 35

[14] Ayan Chakrabarti. 2016. Learning sensor multiplexing design through back-propagation. In *Advances in Neural Information Processing Systems*. 3081–3089. 79

[15] A. Chakrabarti and T. Zickler. 2011. Statistics of Real-World Hyperspectral Images. In *IEEE Conf. Computer Vision and Pattern Recognition*. 34, 98, 100, 107, 108

[16] Huaijin G Chen, Suren Jayasuriya, Jiyue Yang, Judy Stephen, Sriram Sivaramakrishnan, Ashok Veeraraghavan, and Alyosha Molnar. 2016. ASP vision: Optically computing the first layer of convolutional neural networks using angle sensitive pixels. In *IEEE Conf. Computer Vision and Pattern Recognition*. 74, 79

[17] Yushi Chen, Hanlu Jiang, Chunyang Li, Xiuping Jia, and Pedram Ghamisi. 2016. Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *Trans. Geoscience and Remote Sensing* 54, 10 (2016), 6232–6251. 73, 74

[18] Inchang Choi, Daniel S. Jeon, Giljoo Nam, Diego Gutierrez, and Min H. Kim. 2017. High-Quality Hyperspectral Reconstruction Using a Spectral Prior. *ACM Trans. Graphics* 36, 6 (2017), 218:1–13. 5, 35, 50, 51, 98, 99, 100, 103, 107

[19] EA Cloutis. 1996. Review Article Hyperspectral geological remote sensing: evaluation of analytical techniques. *International J. Remote Sensing* 17, 12 (1996), 2215–2242. 13, 31, 69

[20] Norman Colthup. 2012. *Introduction to infrared and Raman spectroscopy*. Elsevier. 13, 69

[21] Oliver Cossairt, Mohit Gupta, and Shree K Nayar. 2012. When does computational imaging improve performance? *IEEE Trans. Image Processing* 22, 2 (2012), 447–458. 3

[22] Michael G Cowling and John F Price. 1984. Bandwidth versus time concentration: the Heisenberg–Pauli–Weyl inequality. *SIAM J. Mathematical Analysis* 15, 1 (1984), 151–165. 17

[23] Mark A Davenport, Jason N Laska, John R Treichler, and Richard G Baraniuk. 2012. The pros and cons of compressive sensing for wideband signal acquisition: Noise folding versus dynamic range. *IEEE Trans. Signal Processing* 60, 9 (2012), 4628–4642. 4

[24] Nicolas Dobigeon, Jean-Yves Tourneret, Cédric Richard, José Carlos M Bermudez, Stephen McLaughlin, and Alfred O Hero. 2014. Nonlinear unmixing of hyperspectral images: Models and algorithms. *IEEE Signal Processing Magazine* 31, 1 (2014), 82–94. 72

[25] Zackory Erickson, Nathan Luskey, Sonia Chernova, and Charles C Kemp. 2019. Classification of household materials via spectroscopy. *IEEE Robotics and Automation Letters* 4, 2 (2019), 700–707. 72, 81

[26] Mathieu Fauvel, Jocelyn Chanussot, Jon Atli Benediktsson, and Johannes R Sveinsson. 2007. Spectral and spatial classification of hyperspectral data using SVMs and morphological profiles. In *IEEE Int. Geoscience and Remote Sensing Symposium*. 4834–4837. 73

[27] M Fazel, E Candes, B Recht, and P Parrilo. 2008. Compressed sensing and robust recovery of low rank matrices. In *Asilomar Conf. Signals, Systems and Computers*. 35, 52, 53, 98, 99, 103

[28] Graham D Finlayson, Mark S Drew, and Brian V Funt. 1994. Color constancy: generalized diagonal transforms suffice. *JOSA A* 11, 11 (1994), 3011–3019. 34

[29] Hongya Ge, Ivars P Kirsteins, and Louis L Scharf. 2006. Data dimension reduction using Krylov subspaces: Making adaptive beamformers robust to model order-determination. In *Intl. Conf. Acoustics Speech and Signal Processing (ICASSP)*. 36

[30] Hongya Ge, LL Scharf, and Magnus Lundberg. 2004. Reduced-rank multiuser detectors based on vector and matrix conjugate gradient Wiener filters. In *Workshop on Signal Processing Adv. Wireless Communications*. 36

[31] ME Gehm, R John, DJ Brady, RM Willett, and TJ Schulz. 2007. Single-shot compressive spectral imaging with a dual-disperser architecture. *Optics Express* 15, 21 (2007), 14013–14027. 35

[32] Mayank Goel, Eric Whitmire, Alex Mariakakis, T Scott Saponas, Neel Joshi, Dan Morris, Brian Guenter, Marcel Gavriliu, Gaetano Borriello, and Shwetak N Patel. 2015. HyperCam: hyperspectral imaging for ubiquitous computing applications. In *ACM Intl. Joint Conf. Pervasive and Ubiquitous Computing*. 145–156. 74

[33] Mohammad Golbabaee and Pierre Vandergheynst. 2012. Hyperspectral image compressed sensing via low-rank and joint-sparse matrix recovery. In *ICASSP*. 34

[34] Gene Golub and William Kahan. 1965. Calculating the singular values and pseudo-inverse of a matrix. *J. of the Society for Industrial and Appl. Mathematics, Series B: Numerical Analysis* 2, 2 (1965), 205–224. 32, 36

[35] Joseph W Goodman. 2005. *Introduction to Fourier optics*. Roberts and Company Publishers. 17, 18, 40

[36] Ali Grami. 2016. Chapter 3 - Signals, Systems, and Spectral Analysis. In *Introduction to Digital Communications*. 41 – 150. 17

[37] Ralph Gross, Iain Matthews, and Simon Baker. 2002. Fisher light-fields for face recognition across pose and illumination. In *Joint Pattern Recognition Symposium*. 481–489. 74

[38] Amina Ben Hamida, Alexandre Benoit, Patrick Lambert, and Chokri Ben Amar. 2018. 3-D Deep learning approach for remote sensing image classification. *Trans. Geoscience and Remote Sensing* 56, 8 (2018), 4420–4434. 73, 74

[39] Joseph C Harsanyi and C-I Chang. 1994. Hyperspectral image classification and dimensionality reduction: An orthogonal subspace projection approach. *IEEE Trans. Geoscience and Remote Sensing* 32, 4 (1994), 779–785. 31, 69

[40] Martin Harwit and Neil J Sloane. 1979. *Hadamard transform optics*. 34

[41] Miloš Hašan, Fabio Pellacini, and Kavita Bala. 2007. Matrix row-column sampling for the many-light problem. *ACM Trans. Graphics* 26, 3 (2007), 26. 66

[42] Mingyi He, Bo Li, and Huahui Chen. 2017. Multi-scale 3D deep convolutional neural network for hyperspectral image classification. In *Intl. Conf. Image Processing*. 73, 74

[43] V Hernandez, JE Roman, A Tomas, and V Vidal. 2007. Restarted Lanczos bidiagonalization for the SVD in SLEPc. *STR-8, Tech. Rep.* (2007). 36

[44] Wei Hu, Yangyu Huang, Li Wei, Fan Zhang, and Hengchao Li. 2015. Deep convolutional neural networks for hyperspectral image classification. *Journal of Sensors* 2015 (2015). 73

[45] Zhuo Hui, Kalyan Sunkavalli, Sunil Hadap, and Aswin C. Sankaranarayanan. 2018. Illuminant Spectra-based Source Separation using Flash Photography. In *IEEE Intl. Conf. Computer Vision and Pattern Recognition (CVPR)*. 34

[46] Daniel S Jeon, Inchang Choi, and Min H Kim. 2016. Multisampling compressive video spectroscopy. In *Computer Graphics Forum*. 35

[47] Isaac Kauvar, Samuel J Yang, Liang Shi, Ian McDowall, and Gordon Wetzstein. 2015. Adaptive color display via perceptually-driven factored spectral projection. *ACM Trans. Graphics (TOG)* 34, 6 (2015), 165–1. 34

[48] Min H Kim, Todd Alan Harvey, David S Kittle, Holly Rushmeier, Julie Dorsey, Richard O Prum, and David J Brady. 2012. 3D imaging spectroscopy for measuring hyperspectral patterns on solid objects. *ACM Trans. Graphics (TOG)* 31, 4 (2012), 38. 31, 69

[49] David Kittle, Kerkil Choi, Ashwin Wagadarikar, and David J Brady. 2010. Multiframe image estimation for coded aperture snapshot spectral imagers. *Appl. optics* 49, 36 (2010), 6824–6833. 4, 15, 35, 52, 99, 100, 103, 105

[50] BVK Vijaya Kumar and David Casasent. 1981. Eigenvector determination by iterative optical methods. *Applied optics* 20, 21 (1981), 3707–3710. 37

[51] Hyungtae Lee and Heesung Kwon. 2016. Contextual deep CNN based hyperspectral classification. In *Intl. Geoscience and Remote Sensing Symposium*. 73, 74

[52] Te-Won Lee, Thomas Wachtler, and Terrence J Sejnowski. 2000. The spectral independent components of natural scenes. In *Intl. Workshop on Biologically Motivated Comp. Vision*. 31

[53] Anat Levin, Rob Fergus, Frédo Durand, and William T Freeman. 2007. Image and depth from a conventional camera with a coded aperture. *ACM transactions on graphics (TOG)* 26, 3 (2007), 70. 43

[54] Chengbo Li, Ting Sun, Kevin F Kelly, and Yin Zhang. 2012. A compressive sensing and unmixing scheme for hyperspectral data processing. *IEEE Trans. Image Processing* 21, 3 (2012), 1200–1210. 4, 34, 35, 74

[55] Ying Li, Haokui Zhang, and Qiang Shen. 2017. Spectral–spatial classification of hyperspectral imagery with 3D convolutional neural network. *Remote Sensing* 9, 1 (2017), 67. 73, 74

[56] Jeff W Lichtman and José-Angel Conchello. 2005. Fluorescence microscopy. *Nature methods* 2, 12 (2005), 910. 13, 16, 69

[57] Xing Lin, Yebin Liu, Jiamin Wu, and Qionghai Dai. 2014. Spatial-spectral encoded compressive hyperspectral imaging. *ACM Trans. Graphics* 33, 6 (2014), 233. 4, 5, 35, 50, 51, 52, 99, 100, 107

[58] Xing Lin, Gordon Wetzstein, Yebin Liu, and Qionghai Dai. 2014. Dual-coded compressive hyperspectral imaging. *Optics letters* 39, 7 (2014), 2044–2047. 4, 7, 13, 15, 16, 17, 35

[59] S. G. Lipson, H. Lipson, and D. S. Tannhauser. 1995. *Optical Physics*. 22

[60] Bing Liu, Xuchu Yu, Pengqiang Zhang, Xiong Tan, Anzhu Yu, and Zhixiang Xue. 2017. A semi-supervised convolutional neural network for hyperspectral image classification. *Remote Sensing Letters* 8, 9 (2017), 839–848. 73, 74

[61] Chao Liu and Jinwei Gu. 2014. Discriminative illumination: Per-pixel classification of raw materials based on optimal projections of spectral brdf. *IEEE trans. pattern analysis and machine intelligence* 36, 1 (2014), 86–98. 74

[62] Adolf W Lohmann, Rainer G Dorsch, David Mendlovic, Zeev Zalevsky, and Carlos Ferreira. 1996. Space–Bandwidth Product of Optical Signals and Systems. *J. Optical Society of America A* 13, 3 (1996), 470–473. 17

[63] Steven P Love and David L Graff. 2014. Full-frame programmable spectral filters based on micromirror arrays. *J. Micro/Nanolithography, MEMS, and MOEMS* 13, 1 (2014), 011108. 7, 13, 15, 16, 17, 18, 29, 75

[64] Yanan Luo, Jie Zou, Chengfei Yao, Xiaosong Zhao, Tao Li, and Gang Bai. 2018. HSI-CNN: A Novel Convolution Neural Network for Hyperspectral Image. In *Intl. Conf. Audio, Language and Image Processing*. 73, 74

[65] Farid Melgani and Lorenzo Bruzzone. 2004. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans. geoscience and remote sensing* 42, 8 (2004), 1778–1790. 73

[66] Ankit Mohan, Ramesh Raskar, and Jack Tumblin. 2008. Agile spectrum imaging: Programmable wavelength modulation for cameras and projectors. In *Computer Graphics Forum*. 7, 13, 16, 17, 34, 75

[67] Shree K Nayar and Ruud M Bolle. 1993. Computing reflectance ratios from an image. *Pattern recognition* 26, 10 (1993), 1529–1542. 86

[68] Shree K Nayar and Ruud M Bolle. 1993. Reflectance ratio: A photometric invariant for object recognition. In *International Conference on Computer Vision*.

[69] Shree K Nayar and Ruud M Bolle. 1996. Reflectance based object recognition. *Intl. J. of Comp. Vision* 17, 3 (1996), 219–240. 86

[70] Matthew O'Toole and Kiriakos N Kutulakos. 2010. Optical computing for fast light transport analysis. *ACM Trans. Graph.* 29, 6 (2010), 164. 37, 74

[71] Jiawei Ou and Fabio Pellacini. 2011. LightSlice: matrix slice sampling for the many-lights problem. *ACM Trans. Graphics* 30, 6 (2011), 179–1. 66

[72] Zhihong Pan, Glenn Healey, Manish Prasad, and Bruce Tromberg. 2003. Face recognition in hyperspectral images. *IEEE Trans. Pattern Analysis and Machine Intelligence* 25, 12 (2003), 1552–1560. 31, 69

[73] Jong-Il Park, Moon-Hyun Lee, Michael D Grossberg, and Shree K Nayar. 2007. Multispectral imaging using multiplexed illumination. In *IEEE Intl. Conf. Computer Vision*. 6, 74

[74] Jussi PS Parkkinen, Jarmo Hallikainen, and Timo Jaaskelainen. 1989. Characteristic spectra of Munsell colors. *JOSA A* 6, 2 (1989), 318–322. 6, 7, 31

[75] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in PyTorch. In *NIPS-W*. 80, 85

[76] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830. 80, 85

[77] Rocco Pierri, Angelo Liseno, Raffaele Solimene, and Francesco Tartaglione. 2002. In-depth resolution from multifrequency Born fields scattered by a dielectric strip in the Fresnel zone. *J. Optical Society of America A* 19, 6 (2002), 1234–1238. 17

[78] Henri Rajbenbach, Yeshayahu Fainman, and Sing H Lee. 1987. Optical implementation of an iterative algorithm for matrix inversion. *Appl. optics* 26, 6 (1987), 1024–1031. 37

[79] Ana Ramirez, Gonzalo R Arce, and Brian M Sadler. 2012. Coded-aperture compressive spectral image classification. In *Computational Optical Sensing and Imaging*. Optical Society of America, CM4B–6. 5, 74

[80] Ana Ramirez, Gonzalo R Arce, and Brian M Sadler. 2012. Hyperspectral pixel classification from coded-aperture compressive imaging. In *Independent Component Analyses, Compressive Sampling, Wavelets, Neural Net, Biosystems, and Nanoengineering X*, Vol. 8401. Intl. Society for Optics and Photonics, 84010J.

[81] Ana Ramirez, Henry Arguello, Gonzalo R Arce, and Brian M Sadler. 2013. Spectral image classification from optimal coded-aperture compressive measurements. *IEEE Trans. Geoscience and Remote Sensing* 52, 6 (2013), 3299–3309. 5, 74

[82] Ramesh Raskar, Amit Agrawal, and Jack Tumblin. 2006. Coded exposure photography: motion deblurring using fluttered shutter. In *ACM Trans. Graphics (TOG)*. 43, 44

[83] Behnood Rasti, Johannes R Sveinsson, Magnus O Ulfarsson, and Jon Atli Benediktsson. 2013. Hyperspectral image denoising using a new linear model and sparse regularization. In *Geoscience and Remote Sensing Symposium, IEEE Intl.* 34

[84] Hoover Rueda, Henry Arguello, and Gonzalo R Arce. 2016. Compressive spectral testbed imaging system based on thin-film color-patterned filter arrays. *Appl. Optics* 55, 33 (2016), 9584–9593. 36

[85] Hoover Rueda, Henry Arguello, and Gonzalo R Arce. 2017. High-dimensional optimization of color coded apertures for compressive spectral cameras. In *European Signal Processing Conf.* 36

[86] Neda Salamati, Clément Fredembach, and Sabine Süsstrunk. 2009. Material classification using color and NIR images. In *Color and Imaging Conference*. 81

[87] Vishwanath Saragadam and Aswin C Sankaranarayanan. 2019. KRISM—Krylov Subspace-based Optical Computing of Hyperspectral Images. *ACM Trans. Graphics* 38, 5 (2019), 148:1–14. 13, 15, 16, 18, 29, 74, 75, 76, 84

[88] Vishwanath Saragadam and Aswin C Sankaranarayanan. 2019. Programmable Spectrometry–Per-pixel Classification of Materials using Learned Spectral Filters. *arXiv preprint arXiv:1905.04815* (2019). 13, 16, 29

[89] Vishwanath Saragadam, Jian Wang, Xin Li, and Aswin Sankaranarayanan. 2017. Compressive spectral anomaly detection. In *Intl. Conf. Comp. Photography*. 2, 34

[90] Vivek Sharma, Ali Diba, Tinne Tuytelaars, and Luc Van Gool. 2016. Hyperspectral CNN for image classification & band selection, with application to face recognition. *Technical report KUL/ESAT/PSI/1604, KU Leuven, ESAT, Leuven, Belgium* (2016). 73, 74

[91] Valeria Simoncini and Daniel B Szyld. 2003. Theory of inexact Krylov subspace methods and applications to scientific computing. *SIAM J. Scientific Computing* 25, 2 (2003), 454–477. 67

[92] SpecTIR. 2019. SpecTIR, Advanced Hyperspectral and Geospatial Solutions. `http://www.spectir.com/free-data-samples/`. [Online; accessed: 2019-01-07]. 99, 104

[93] Ting Sun and Kevin Kelly. 2009. Compressive sensing hyperspectral imager. In *Computational Optical Sensing and Imaging*. 34, 35, 38, 52, 99, 103, 105

[94] Tsuyoshi Takatani, Takahito Aoto, and Yasuhiro Mukaigawa. 2017. One-shot Hyperspectral Imaging using Faced Reflectors. In *CVPR*. 35

[95] Yuliya Tarabalka, Jocelyn Chanussot, and Jon Atli Benediktsson. 2010. Segmentation and classification of hyperspectral images using watershed transformation. *Pattern Recognition* 43, 7 (2010), 2367–2379. 31, 69

[96] Zhi Tian, Hongya Ge, and Louis L Scharf. 2005. Low-complexity multiuser detection and reduced-rank Wiener filters for ultra-wideband multiple access. In *Intl. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*. 36

[97] Ashok Veeraraghavan, Ramesh Raskar, Amit Agrawal, Ankit Mohan, and Jack Tumblin. 2007. Dappled photography: Mask enhanced cameras for heterodyned light fields and coded aperture refocusing. *ACM Trans. Graph.* 26, 3 (2007), 69. 43

[98] Ashwin Wagadarikar, Renu John, Rebecca Willett, and David Brady. 2008. Single disperser design for coded aperture snapshot spectral imaging. *Appl. Optics* 47, 10 (2008), B44–B51. 4, 15, 35, 50

[99] Andrew E Waters, Aswin C Sankaranarayanan, and Richard Baraniuk. 2011. SpaRCS: Recovering low-rank and sparse matrices from compressive measurements. In *Adv. Neural Info. Processing Systems*. 34

[100] Wikipedia. 2019. Liquid Crystal Tunable Filter. https://en.wikipedia.org/wiki/Liquid_crystal_tunable_filter. [Online; accessed: 2019-07-18]. 16

[101] Michael E. Winter. 1999. N-FINDR: An algorithm for fast autonomous spectral end-member determination in hyperspectral data. 34

[102] Fumihito Yasuma, Tomoo Mitsunaga, Daisuke Iso, and Shree K Nayar. 2010. Generalized assorted pixel camera: postcapture control of resolution, dynamic range, and spectrum. *IEEE trans. on image processing* 19, 9 (2010), 2241–2253. 98, 100

[103] Roberta H Yuhas, Alexander FH Goetz, and Joe W Boardman. 1992. Discrimination among semi-arid landscape endmembers using the spectral angle mapper (SAM) algorithm. (1992). 62

[104] Zhengyun Zhang and Marc Levoy. 2009. Wigner Distributions and how they Relate to the Light Field. In *IEEE Intl. Conf. Computational Photography (ICCP)*. 17

[105] Yong-Qiang Zhao and Jingxiang Yang. 2015. Hyperspectral image denoising via sparse representation and low-rank constraint. *IEEE Trans. Geoscience and Remote Sensing* 53, 1 (2015), 296–308. 34

[106] Tiancheng Zhi, Bernardo R Pires, Martial Hebert, and Srinivasa G Narasimhan. 2019. Multispectral Imaging for Fine-Grained Recognition of Powders on Complex Backgrounds. In *IEEE Intl. Conf. Comp. Vision and Pattern Recognition (CVPR)*. 13, 16