Programmable Spectral Filter Arrays for Hyperspectral Imaging

ASWIN C. SANKARANARAYANAN, ECE Department, Carnegie Mellon University, USA
VISHWANATH SARAGADAM, ECE Department, Rice University, USA
VIJAY RENGARAJAN, ECE Department, Carnegie Mellon University, USA
RYUICHI TADANO, TUO ZHUANG, HIDEKI OYAIZU, and JUN MURAYAMA, Sony Group Corporation, Japan

1 INTRODUCTION

A basic tool in the repertoire of hyperspectral imaging is the ability to sample the incident light in space as well as spectrum. Sampling the scene along the spatial dimension is straightforward, achieved by focusing light using a lens onto a sensor. However, doing so while simultaneously resolving spectrum at high resolution is challenging, in part due to the paucity of light, when resolved into narrowband spectra, and in part due to the dimensionality of the resulting signal, which requires a large amount of time to sample and digitize. Circumventing these challenges requires novel imaging systems that can probe the spatio-spectral dimensions of light in rich ways.

There are numerous approaches for performing spectral modulation in the context of hyperspectral imaging. Perhaps the simplest approach is to use a set of color filters, by sequentially scanning them one at a time. However, this approach scales poorly when we seek to resolve spectrum at high resolution. A more popular approach in optics is to spectrally resolve light using the dispersive property of a prism or a diffraction grating, to form the so-called rainbow plane [Mohan et al. 2008], an image where different wavebands of light are focused at different spatial locations. The resulting spectral image allows us to modulate the spectral content of the scene, a technique that has been used for hyperspectral imaging [Lin et al. 2014] as well as spectral filtering [Mohan et al. 2008]. Liquid crystal (LC) cells offer a different approach where-in the birefringence of the material produces wavelength-dependent modulation [Wu et al. 1984]; changing the voltage across the LC cell changes the birefringence of the device, which in turn, changes the spectral response of the device. LC cells can be used to create sinusoidal as well as narrowband filters by stacking multiple cells and has found application in estimating freshness of fruits [Peng and Lu 2006], improving contrast in bioimaging [Gebhart et al. 2007], and detection of vegetation [Li et al. 2018] and powders [Zhi et al. 2019]. For the most part, all of these techniques modulate the spectral dimension of light across the entire scene without fine-grain spatial selectivity.

Modulating the spectral dimension of light has numerous applications in computational imaging. While there are many techniques for achieving this, there are few, if any, for implementing a spatially-varying and programmable spectral filter. This paper provides an optical design for implementing such a capability. Our key insight is that spatially-varying spectral modulation can be implemented using a liquid crystal spatial light modulator since it provides an array of liquid crystal cells, each of which can be purposed to act as a programmable spectral filter array. Relying on this insight, we provide an optical schematic and an associated lab prototype for realizing the capability, as well as address the associated challenges at implementation using optical and computational innovations. We show a number of unique operating points with our prototype including single- and multi-image hyperspectral imaging, as well as its application in material identification.

CCS Concepts: • Computing methodologies → Hyperspectral imaging; Computational photography.

Additional Key Words and Phrases: spectral modulation, liquid crystal cells, phase spatial light modulators

Fig. 1. We propose a computational camera that implements a spatially-varying programmable spectral filter array. Our camera can capture intensity images under a wide configuration of spectral filters. Subsequently, we can computationally recover the hyperspectral image of the scene from one or more such spectrally filtered images. Shown above are the results of a butterfly recovered with eight spectrally-filtered images.
There are few, if any, approaches that provide programmable spatio-spectral modulation, i.e., the ability to independently modulate the spectrum of light at different spatial locations.

This paper provides the design of an optical system that implements a programmable spectral filter array, i.e., a system that, within limits, provides the capability to choose freely among a selection of spectral filters at each pixel of the sensor. Our key insight is such a system can be realized by using a liquid crystal-based spectral light modulator (SLM). Each pixel of the SLM acts as an independent LC cell which can realize a set of spectral filters depending on the applied voltage. By optically aligning an image sensor to the SLM, we can leverage the degrees of freedom provided by the millions of independently-addressable pixels of the SLM to implement a programmable spatially-varying spectral filter.

There are numerous benefits to be derived from a programmable spectral filter array. We can implement a single-shot hyperspectral imaging technique, simply by spatially tiling the spectral filters. However, unlike color filter arrays which are fixed at fabrication, we can dynamically change the filter arrays to enable a framework for few-shot hyperspectral imaging, as seen in Figure 1, or to tailor the tiling to be application specific, for example, using a set of filters to disambiguate between different materials.

Contributions. This paper proposes an optical system for implementing a programmable spectral filter array for sensing hyperspectral images (HSIs), making the following contributions.

- **Programmable spectral filter array using phase-only SLMs.** We provide a novel design for implementing an LC-based filter array by using an SLM that is optically aligned to an image sensor.
- **Reducing the effect of phase modulation.** Implementing the desired filter with a phase SLM introduces a spatially-varying phase retardation, which has undesirable effects in the form of angular tilts. We minimize these effects by a careful design of the patterns displayed on the SLM. We also train a deep network to correct for unmodeled aberrations introduced by the optical setup.
- **Dataset.** We introduce a hyperspectral image dataset, acquired using our lab prototype. The dataset comprises of scenes under a variety of spectral filters, including a Nyquist scan captured with spatially-constant filter, sequentially scanned one at a time, as well as a range of spatially-varying patterns.
- **Hyperspectral image reconstruction.** Finally, we reconstruct HSIs from single as well as multiple image measurements, each captured using a different spectral filter array.

Limitations. We inherit a number of limitations due to our use of phase SLMs and LC cells for spectral modulation. Spectral filters implemented using individual LC cells are typically restricted to sinusoidal profiles, with a spectral resolution that is inversely proportional to wavelength. Further, phase SLMs can only implement spatial patterns that are smooth, which limits the range of filters we can implement. Another limitation stems from the shallow range of phase retardation in commercial phase SLMs, which in turn limits the number of distinct spectral filters that we can implement, and consequently, the spectral resolution that we can achieve.

2 PRIOR WORK

We briefly discuss key results in spectral modulation techniques, and their use in hyperspectral imaging.

2.1 Compressive Hyperspectral Imaging

Classic techniques for hyperspectral imaging, such as the pushbroom camera and tunable filters, perform a time-consuming sequential scan along the spatial or spectral dimensions. One way to speed up the acquisition is a class of techniques, referred to as compressive sensing (CS), where lower-dimensional signal models for HSIs are leveraged to reconstruct the HSI from an underdetermined set of linear measurements. There are many such signal models that apply to HSIs. Since natural images enjoy sparsity in transform domain as well as spatial gradients, HSIs are well represented by group sparsity across the spectral channels as well as cross-channel gradient priors. Other popular choices include modeling the HSI as a low-rank matrix or a sum of low-rank and sparse matrices. Immaterial of the model used, CS relies on taking coded linear measurements and solving a linear inverse problem, regularized by the chosen low-dimensional representation. The ideas of CS and its associated reconstruction techniques apply to the proposed system as its measurement operator is both linear and under-determined.

There have been many imaging architectures proposed to enable compressive sensing of HSIs. The coded aperture snapshot spectral imaging (CASSI) architecture [Gehm et al. 2007; Wagadarikar et al. 2008] modifies the pushbroom camera by replacing the slit on the image plane with a random 2D binary mask. The captured image forms an underdetermined set of linear measurements from which the HSI is computationally reconstructed.

Compressive hyperspectral imaging has benefited from many interesting optical architectures. Lin et al. [2014] perform a spatial coding followed by spectral coding, before capturing an RGB image of the scene. Within each exposure of the sensor, multiple spatial-spectral patterns are used to create a rich spatially-varying spectral filter; in this regard, this technique is similar to the proposed approach. However, a limitation of time multiplexed measurements is the severe loss in light levels by an amount equal to the number of patterns used within each exposure. Sun and Kelly [2009] perform spatial coding on the HSI using a digital micromirror device (DMD), and subsequently use a spectrometer to get the spectral profile of the coded HSI. Baek et al. [2017] place a prism in front of the lens of a camera, which introduces spectral blur on the captured image, from which the HSI is reconstructed using a deep neural network.

The spectral modulation property of LC cells have been used for compressive sensing of HSIs. Oiknine et al. [2019] and August and Stern [2013] capture multiple images of a scene, each with a different but spatially invariant spectral filter, and reconstruct the HSI using traditional transform domain sparsity. The main drawback of this approach is the number of images needed to recover the HSI. The proposed work is a direct extension of this body of work, where we use a spatially-varying counterpart of LC-based spectral filters to provide significant reduction in the number of images needed. There has been some work that uses an LC SLM for spectral imaging [Tsai et al. 2015; Zhu et al. 2013], but none of them use the spatial
programmability of the SLM to create a spatially-varying filter, which forms the main contribution of this work.

2.2 Assorted Pixels

Assorted pixels [Narasimhan and Nayar 2005] refers to a technique where a grayscale image sensor is augmented by placing an array of filters on top to provide enhanced perception of spectrum, polarization and/or dynamic range. Assorted pixels have been extended to hyperspectral imaging as well, where a narrowband spectral filter array is tiled on top of the sensor [Lambrechts et al. 2014]. This approach can be interpreted as an extension of the commonly used Bayer patterns, or color filter arrays [Bayer 1976]. While spectral filter arrays do speedup the acquisition time down to a single image, there is an inherent loss in spatial resolution due to the tiling of the filter array, which can be quite significant when we seek to resolve at high spectral resolutions, over a large waveband. The main contribution of this paper is in the spirit of assorted pixels; but unlike existing work, where the spectral arrays are fixed at fabrication, the proposed optical design permits a programmable array of spectral filters which enhances the scope of the technique in many interesting ways.

3 PROGRAMMABLE SPECTRAL FILTER ARRAYS

We now discuss the core ideas underlying this paper, namely, spectral filtering with LC cells and the implementation of spatially-varying spectral filters using a phase SLM.

3.1 Basics of Spectral Filtering with LC Cells

We briefly go over the principle of operation of an LC cell when used to implement a spectral filter. The reader is referred to [Wu et al. 1984] as well as the Appendix for a detailed treatment.

Consider an imaging setup consisting of an LC cell of thickness $d_{LC}$ that is sandwiched between two linear cross polarizers, with their polarization axes oriented at ±45° to the LC cell’s fast axis. Suppose that we apply a voltage $v$ across the LC cell which produces a birefringence of $\Delta n(v)$. Now, unpolarized light incident on this setup experiences a spectral filter of the form:

$$\frac{1}{2} \left( 1 - \cos \left( 2\pi \frac{\Delta n(v)d_{LC}}{\lambda} \right) \right),$$

(1)

where $\lambda$ is the wavelength of light. This filter is sinusoidal in the wavenumber, or the reciprocal of the wavelength. The frequency of this sinusoid is determined by the term $\Delta n(v)d_{LC}$, which quantizes the path difference introduced by the LC cell. As is to be expected, thicker LC cells introduce a large path difference which creates spectral filters with more cycles over $\lambda$. Similarly, higher levels of birefringence $\Delta n$, that typically happens for low values of $v$, also creates a larger number of cycles over the waveband of interest.

3.2 Programmable Spectral Arrays using SLMs

We now describe the basic ideas underlying the proposed programmable spectral filter array. Our key insight is that an LC SLM

$$\text{in essence an array of LC cells, each of which acts as a programmable spectral filter. Collocating the SLM with an image sensor, hence, allows us to obtain a device whose spectral response can be changed spatially, up to the limits imposed by the device construction. Figure 24 shows the optical schematic such a device, where the SLM and an image sensor are optically collocated to the image plane of a main lens. The polarizing beamsplitter between the SLM and the camera acts as a pair of cross-polarizers.}

For sake of simplicity, let us assume that the SLM pixel size is identical to that of the sensor and that the image relay provides a one-to-one mapping between them. Suppose we display a spatially-varying voltage pattern $v(x,y)$ on the SLM, then the image $i(x,y)$ observed at the sensor is given as

$$i(x,y) = \frac{1}{2} \int_{\lambda} h(x,y,\lambda) \left( 1 - \cos \left( 2\pi \frac{\Delta n(v(x,y)d_{LC}}{\lambda} \right) \right) s(\lambda) d\lambda,$$

(2)

where $s(\lambda)$ is the spectral sensitivity of the image sensor and $h(x,y,\lambda)$ is the unmodulated hyperspectral image formed on the sensor. Hence, by appropriate choice of the pattern that we display on the SLM, we can implement spatially-varying spectral filters. The set of filters we can obtain depends on the birefringence of the SLM, the range of input voltage we can provide, and the thickness of the SLM, all of which are device specific.

3.3 Hardware Prototype

Figure 3 shows the lab prototype, on an optical benchtop, implementing the programmable spectral filter array. For the most part, the prototype corresponds to the optical schematic presented earlier in Figure 24. However, there are some important differences between the prototype and the schematic. First, the fast and slow axes of the SLM are typically aligned to it edges; since we need the incident polarization to be at 45° to the fast axis, we would need to mount the SLM with an in-plane rotation. This, while feasible, is cumbersome. We instead use an achromatic quarterwave plate, immediately after the polarizing beamsplitter, with its fast axis aligned at 45° to the transmitted polarization state of the beamsplitter. The resulting light is circularly polarized, which has equal energy along
both the fast and slow axes of the SLM, and hence, identical to rotating the SLM. Second, we use a camera lens, focused at infinity, in the relay between the SLM and the main camera; the superior multi-lens design of the camera lens has the effect of providing a dramatic improvement in the quality of captured images. Finally, we have a second image sensor, an RGB unit, placed in the previously unused arm of the beamsplitter, and collocated with the image plane of the system and, hence, the SLM and the other image sensor. We refer to this as the guide camera, and use it for guided filter-based reconstruction techniques that resolve the loss of spatial resolution due to tiling of the spectral filters.

Specifications of the system. The system is configured for a waveband spanning visible (VIS) and near-infrared (NIR) wavelengths, i.e., 400–1000nm, which roughly matches the sensor response of the main camera that was used for making the spatio-spectral measurements. We used a Holoeye Pluto2 NIR-015 SLM optimized for NIR wavebands, and endowed with a higher phase retardation which in turn provided a set of spectral filters with larger number of oscillations. The SLM resolution was 1920 × 1080, with a pixel pitch of 8 \( \mu \text{m} \). The pixel pitch of the camera observing the SLM is 6.5 \( \mu \text{m} \), at a resolution of 2048 × 2048 pixels.

The relay lenses used, except for the one immediately in front of the camera(s), were achromatic doublets, optimized for NIR transmissions, which we observed provided a smaller spot size over the VIS-NIR wavebands. However, the use of doublets does lead to a chromatic blur in our captured images, which affects the captured imagery. Clearly, using better optics, perhaps apochromatic lenses optimized to provide better chromatic correction, would naturally lead to better results. The lab prototype above had a spatial field of size approximately 10.4 × 8 sq.mm; this field size, that is smaller than the dimensions of the SLM or camera, allowed us to avoid strong spherical and chromatic aberrations at its periphery. We also set the Fourier plane aperture to be 10mm wide; with 100mm relay lens, this resulted in a system with an f/10 aperture.

Calibration. There are two critical calibration procedures that we need for operating this system. The first is that of spectral calibration, where we obtain the spectral filter implemented for a specific value displayed on the SLM. To do this, we mount a spectrometer (OceanOptics FLAME-S-VIS-NIR-ES) in place of the main camera and sweep 256 constant-valued patterns on the SLM, corresponding to its 8-bit input. For each displayed pattern, we capture a spectral measurement over the range [380, 1010] nm at a spectral resolution of 0.35 nm. The specific filters we obtain depend critically on the gamma curve used in the control of the SLM; we will discuss this in detail in the next section, where we also visualize the set of filters that we obtain with our setup.

We also need to obtain a pixel-to-pixel mapping between the SLM and the spectral camera; this step is especially important since our object is to implement a filter array and hence, this XY calibration provides critical information on the filter applied to obtain the measurement at a sensor pixel. We use conventional techniques to obtain this calibration, where we sequentially scan rows and columns on the SLM, detect the corresponding activated row/column on the image sensor, and finally, robustly fit a polynomial mapping between the two. We provide details of this calibration procedure in the Appendix. Finally, the guide RGB camera is registered first to the spectral camera by imaging a texture-rich scene and fitting a homography between the images captured on the two cameras.

4 DESIGNING SPATIALLY-VARYING FILTERS

We now discuss the choice of spatially-varying pattern that we display on the SLM. Intuitively, the pattern that we display needs to be rich, over local regions, so as to provide a diverse set of spectral filters. However, the complexity of the displayed pattern also needs to be balanced against an unintended consequence of our optical system, namely, distortions to the incident wavefront due to phase modulation induced by the SLM.

Undesired effects of phase modulation. The spatially-varying phase modulation induced by the SLM also distorts the wavefront of light incident on it, in the form of local tilts which are dependent on the spatial gradient of the phase pattern. These tilts can have undesirable effects due to non-idealities in the optical relay between the SLM and the camera. To see why this is the case, consider the light path between the SLM and the image sensor, shown in Figure 4. When there is no phase gradient on the SLM, i.e., a spatially-constant phase, we get the scenario in Figure 4(a), which we observed provided a smaller spot size over the image sensor.

Fig. 4. Vignetting due to phase modulation. As compared to the scenario where a constant phase pattern is shown on the SLM, as seen in (a), a spatially-varying phase modulation distorts the wavefront by introducing local tilts, as seen in (b). This tilt can lead to a portion of the incident light being blocked by the relay lenses.
the relay lenses. As a consequence, the change in intensity that we observe at a sensor pixel, when displaying a spatially-varying pattern on the SLM, ends up being a complex function of spatial location as well as phase gradient. We next look at approaches for reducing these undesirable effects of phase modulation.

4.1 Mitigating the Effect of Phase Modulation

Since the amount of wavefront distortion is directly related to the phase gradient, we begin by deriving the expression for the phase gradient, with the goal of identifying the factors that determine it. From (21), the phase delays $\phi(x, \lambda)$, where $x = (x, y)$, induced by the SLM is given as

$$\phi(x, \lambda) = 2\pi \frac{\Delta n(v(x))d_{LC}}{\lambda}.$$  \hfill (3)

The local distortions to the wavefront can be characterized by the spatial gradients of $\phi$.

In its normal mode of operation, we do not have arbitrary control over the SLM voltage $v(x, y)$ due to constraints on bandwidth of data on the video port used to control the SLM patterns. Instead, the voltage $v(x, y)$ is given as

$$v(x, y) = y(p(x, y)),$$  \hfill (4)

where $p(x, y)$ is the 8-bit image displayed on the video port and $y(\cdot)$ is the “gamma curve” of the SLM, or the mapping of this 8-bit number to voltage applied at the SLM. With this, the expression for the spatial gradients of $\phi(x; \lambda)$ can be written as

$$\nabla \phi = \frac{2\pi}{\lambda} \frac{\Delta n}{d_{LC}} \frac{\partial y}{\partial p} \frac{\partial p}{\partial x}.$$  \hfill (5)

The expression of the gradient above can be broken into three terms:

- $\frac{\partial \Delta n}{\partial p}$: the change in birefringence as a function of voltage across the LC cell — this is a device-specific property,
- $\frac{\partial y}{\partial p}$: gradient of the voltage applied at an SLM pixel as a function of the SLM index value used to control the SLM; this is a term that is determined by the gamma curve that we select, and
- $\frac{\partial p}{\partial x}$: the spatial gradients of the display pattern sent to the SLM.

Our goal is to keep the gradient as small as possible so as to minimize the undesirable effect of light tilts. While it is possible to get a zero-valued gradient simply by displaying a constant pattern $p(x, y) = p$, such a choice is inconsistent with a spectral filter array.

**Designing the gamma curve.** To simplify the overall design problem, while minimizing the effect of the phase gradient, we choose a gamma curve $y(\cdot)$ for the SLM such that

$$\frac{\partial y}{\partial p} = c_0 \left[ d_{LC} \frac{\partial \Delta n}{\partial p} \right]^{-1}.$$  \hfill (6)

where $c_0$ is a constant. This is equivalent to the selecting a $\gamma$ such that $\Delta n(y(p))$ is affine in $p$. The procedure for designing this gamma curve requires knowledge of the phase retardance induced by the SLM at different input voltages; we illustrate the procedure for this in Figure 26 and provide a detailed description in the Appendix. With this choice, the phase gradient in (5) reduces to

$$\nabla \phi = \frac{2\pi c_0}{\lambda} \frac{\partial p}{\partial x},$$  \hfill (7)

and hence, the phase gradient is directly controlled by the smoothness of the pattern we display.

4.2 Designing SLM Patterns

The design of the patterns displayed on the SLM needs to balance two key criteria. At one end, we need a diverse set of spectral filters in any local patch, so that we have a rich measurement operator. This richness criteria is best satisfied with patterns that are endowed with very different filters in immediate proximity. At the other end, we need to ensure that the spatial gradient of the displayed pattern is small; this is necessary so as to avoid the adverse effects of the distortions introduced by the spatially-varying phase modulation. This smoothness criteria is best satisfied by a constant pattern. Clearly, these two criteria are in direct opposition with one another. Armed with these two criteria, we consider a number of patterns, discuss the rationale behind each, and detail their relative merits. This discussion is summarized in Figure 6.

**Linear patterns and staggered variants.** We can introduce some diversity into the SLM pattern by varying it linearly in a single
Fig. 6. Pattern design. We explore a range of SLM patterns that offer varying levels of smoothness, measured in terms of the phase gradient, and local richness, measured in terms of range of unique SLM values in a compact neighborhood. The smoothest of such patterns (not shown in the image above) are constant-valued SLM patterns, where there is no spatial gradient, that lack any diversity in spectral measurements. One-dimensional patterns offer the next smoothest alternative. (col 1,2) We design 1D patterns that are three pixels wide/tall and then stagger them to generate a diverse set of spectral in a local neighborhood. (col 3,4) We can also scale the patterns to generate more spectral filters at the cost of increase phase gradient. (col 5-8) Two-dimensional patterns offer the entire gamut of spectral filters in a small neighborhood, but generally at higher phase gradients than 1D patterns. We explore patterns that are tilted in two different ways: periodic and mirror symmetric, which avoid sharp transitions. (col-9) Finally, we design random tilings of $3 \times 3$ blocks; this provides a rich set of patterns, albeit with high phase gradients that make the measurement unreliable near the boundaries of each block. For each pattern, we show (row-1) the pattern displayed on the SLM, (row-2) a simulated image obtained from a full scan of all spectral filters, (row-3) the measured image, and (row-4) the absolute difference between the simulated and measured image. Note the high levels of inaccuracies near discontinuities in the SLM image.

direction. An example of this pattern is

$$p(x, y) = x \mod 255,$$

which has a spatial gradient of 1 intensity per SLM pixel (we will ignore these units in the sequel). This pattern is a slight improvement over the constant pattern, but has tiles that are stretched along one of the axis, which results in severe loss in spatial resolution. One approach to decrease this loss in resolution is to scale the pattern by displaying

$$p(x, y) = (2x) \mod 255,$$

which increases the gradient by a factor of two, but decreases the tiling size by a commensurate amount. A richer way to balance out this loss of resolution is to stagger the pattern across different rows, as shown in Figure 6. This staggering intentionally introduces vertical discontinuities, whose effects can be minimized simply by repeating the pattern for a few rows and rejecting measurements at the discontinuity. Staggering allows us to avoid suffering a severe loss in resolution along one axis.

Two-dimensional patterns. Smooth two-dimensional patterns can be designed by tiling the 256 spectral filters into a $16 \times 16$ tile. Here, we have multiple choices in the form of the direction of tiling, which can be horizontal, or vertical, as well as in the nature of the tiling, which can be periodic or mirror symmetric. Among these, the spatial gradients for the horizontal and vertical oriented patterns are $240/16 = 15$. Periodic tiling have strong discontinuities at the edge of each tile, but any $16 \times 16$ tile has all the choices of filters that the SLM can offer. In contrast, symmetric tilings have no discontinuities, but do not guarantee that any $16 \times 16$ patch covers all possible filters.

Random patterns. Finally, it is worth considering random tiling of filters, which greatly increases the diversity of filters available in any local patch. To improve the smoothness of such patterns, we can repeat each random pattern in a small local window of $3 \times 3$ or $5 \times 5$ pixels so that the measurements at the center of each window have little aberrations.

4.3 Dataset
We collected a dataset of scans with our lab prototype. The dataset comprised of indoor scenes, comprising mainly of single or multiple objects. Figure 7 provides the RGB image, captured with the guide camera, for each of the scenes. We illuminated the scenes with a number of different sources including an NIR-enhanced incandescent light, a cool white LED, and CFL lamps. All acquired images, both from the grayscale camera and the RGB camera, are registered to the SLM using the calibration. After mapping to the SLM, the
images are cropped to the central 1024 × 1024 pixels. For each scene, we captured two sets of patterns that we describe next.

**Full Scan.** For each scene, we acquire a set of 256 images, corresponding to constant patterns displayed on the SLM, and the 92 spatially-varying patterns as described in Table 1. The scenes marked with a green boundary were used for testing and the rest for training the restoration network.

**Spatially-varying filtered images.** For each scene, we acquire a set of 92 images corresponding to the SLM displaying a constant intensity pattern — one for each of the 8-bit intensity control that we have. Note that this results in a constant voltage for all SLM pixels, via the “gamma” function that we derive in Section 4.1. We refer to this set as the full scan measurements. We use it as the baseline and reconstruct a nominal ground truth hyperspectral image. This set also has none of the aberrations introduced due to phase modulation, since we are displaying a constant pattern on the SLM. Further, we can use this set of 256 images to simulate the ideal measurement that we would obtain when displaying a spatially-varying pattern. We term these as “simulated measurements”; example of such measurements are shown in the second row of Figure 6.

**Spatially-varying patterns.** For each scene, we acquire a set of 92 images corresponding to the SLM displaying the patterns types described in Figure 6. For each type, we display multiple patterns by performing circular shifts of the patterns as appropriate to the type. For example, 1D patterns are shifted in just one dimension, while 2D patterns are perturbed along both dimensions. Random patterns are simply regenerated to get entirely independent images. The list of patterns with an enumeration of how we obtain a set of 92 patterns is provided in Table 1. The third row of Figure 6 illustrates the actual patterns captured for a scene where a diffuser is placed in the image plane of the setup. These measurements have a significant mismatch against the measurements simulated with the full scan data. We visualize these differences in the last row of the figure. The differences are especially significant near discontinuities of the pattern as is clearly seen in the 1D measurements, and extend beyond the discontinuities for the 2D patterns.

The mismatch between the simulated and measured images can be attributed to the following reasons. First, the SLM displays a smoothed version of the pattern we display; hence, we can expect significant mismatch between what we want to display and what the SLM implements at discontinuities. Second, there are aberrations induced due to phase distortions introduced by the SLM when we display a spatially-varying pattern; as described earlier, this leads to pattern specific errors that is prominent at places with large phase gradient. Third, the doublet lenses used in the relays introduced chromatic blur, which can be quite large given our operating range of 450-950nm, a span of wavelengths that covers visible and near-infrared bands. Due to this, a single sensor pixel measures light from multiple SLM pixels, which leads to a corrupted measurement that corresponds to a blurred measurement in both space as well as the spectrum. Note that this chromatic blur is also present in the full scan data; however, since the full scan data is measured with a constant spectral filter, the resulting blur is purely spatial. To reduce all of these non-idealities in measured data, we rely on a learning-based approach using neural networks.

### 4.4 Restoration of Assorted Measurements

Our goal is to learn a mapping that takes the distorted measurements and produce clean “simulated” measurements. Since interactions between neighboring spatial filters are non-linear, and often spatially varying, simple models may not capture the non-idealities. We instead rely on a learning-based approach to produce accurate measurements.

We design a single neural network that takes the measured image as input, for any spatially-varying pattern on the SLM that is also provided as input, and outputs the “simulated” measurement for that pattern produced from the full scan data. To account for the spatially varying artifacts, we include the coordinates \((x, y)\), the pattern index \(p\) to the inputs, in addition to the measured intensity at each pixel. Based on the recent works of [Vaswani et al. 2017] and [Mildenhall et al. 2020] in positional encoding, we transform each of \(x, y,\) and \(p\) into 64-length encoded vectors instead of using their original

---

**Table 1.** Each of the patterns in Fig. 6 under different perturbations. For the non-random patterns, these are circular shifts in one or both directions, as appropriate for the pattern. For the random pattern, we simply regenerate the random patterns. In total, we capture each scene with 92 patterns.

| Name                       | #patterns | perturbation          |
|----------------------------|-----------|-----------------------|
| 1D horizontal              | 16        | horizontal shifts     |
| 1D vertical                | 16        | vertical shifts       |
| 1D horizontal (scale 2x)   | 8         | horizontal shifts     |
| 1D horizontal (scale 4x)   | 4         | horizontal shifts     |
| 2D vertical (periodic)     | 8         | 2D shifts             |
| 2D vertical (mirror)       | 8         | 2D shifts             |
| 2D horizontal (periodic)   | 8         | 2D shifts             |
| 2D horizontal (mirror)     | 8         | 2D shifts             |
| random                     | 16        | random tiles          |
| **Total**                  | **92**    |                       |
Fig. 8. Restoration network architecture. Our pattern-oblivious encoder-decoder architecture rectifies the distortions in the measurement image using positional codes based on coordinates and pattern index.

Fig. 9. Restoration of measured image. We show 128 × 128 patches of a scene from our test data. Each row in the left and right page columns corresponds to some of the different patterns that we employed during capture. The simulated measurement in (a) is the “ground-truth” image for each pattern. The measured image in (b) is very much distorted as can be seen by its absolute difference with the simulated measurement in (c). Our restoration network cleans up the distortions and produces the restored measurement image shown in (d) which has much lower distortions as can be seen in (e). The efficacy of our restoration network can also be seen from much higher reconstruction PSNR values which are shown as insets.

scalar values. Given a pos  ∈ {x, y, p}, we use [cos(pos · f2i/N), sin(pos · f2i/N)], for i = 0, . . . , N/2−1 with N = 64 and f = 10−4 as the 64-length positional encoding vector as suggested by [Vaswani et al. 2017]. Thus, the input to the network consists of a measurement intensity channel, and 64 channels each for x, y and p, resulting in a total of 193 channels, as shown in Figure 8. The output of the network is a single restored image channel corresponding to the simulated measurement.

Network Architecture. We use an encoder-decoder architecture similar to that of U-net proposed by [Ronnebergen et al. 2015] with four downsampling and four upsampling blocks as shown in Figure 8. The specifics of this network and how it was trained is provided in the supplement.

Fig. 10. Improvements from restoration. The restoration network provides measurements that have a higher fidelity to the ideal model, by correcting for blur and vignetting introduced by imprecision in the optical implementation. We can see that, across all 92 SLM patterns, it provides an improvement of 7 dB or higher, as compared to the raw measurements. The numbers shown here are averages, over all the test scenes.

Restoration. During inference, we input the measured image along with the positional encoding channels (all as 1024 × 1024 images) to the trained network to get the restored image as output which will be used for hyperspectral image reconstruction. We show the results of our restoration network for a test scene which is not used in training and validation in Figure 9.

Performance of the restoration strategy. Figure 10 evaluates the performance of the restoration network on a test set of seven scenes. We quantify the error between the simulated and restored measurements, averaged across the seven scenes, for each of the 92 patterns, and compare them to that of the error associated with the measured images. We use peak signal-to-noise ratio (PSNR), measured in dB, as our choice of metric which is defined as:

\[
\text{PSNR} = 20 \log_{10} \left( \frac{\|x\|_\infty}{\text{RMSE}(x; \hat{x})} \right),
\]

where \(x\) and \(\hat{x}\) are the simulated image and the restored/measurement image, respectively, with an intensity range [0–1]. We observe an improvement that is often greater than 7 dB, indicating significant
increase in measurement fidelity to the simulated data once we apply the restoration network.

Restoration of unseen patterns. Since our restoration network is trained with the pattern code as well as an input, it is capable of restoring unseen patterns as well. We show the restoration outputs of patterns, previously unseen during training, in Figure 11. Our network successfully restores these patterns, albeit with a slight loss in performance as compared to the patterns used at training. This indicates that our trained network does not merely memorize a pseudo-random distribution of the patterns present in the training set, but can generalize for any pattern.

5 HYPERSONAL RECONSTRUCTION ALGORITHMS

We propose two reconstruction techniques to obtain HSIs from our imaging setup. The goal of both approaches is to estimate the HSI from single or multiple image measurements, each of which is obtained with a (different) spatially-varying pattern on the SLM. The differences between the two methods stem from whether or not they use an auxiliary RGB image acquired by the guide camera.

5.1 Reconstruction without using Guide Image

The measurements made with our imaging setup are linear in the unknown HSI, and hence, reconstruction can be posed as a simple linear inverse problem. Let $H[x, y, l]$ be the discretized version of the HSI with $[x, y]$ representing discrete spatial coordinates and $l$ representing discrete wavelength coordinates. Let $\Phi_k[x, y, l]$ be the spatially varying set of spectral filters for pattern $k$. The set of measurements is then given by

$$I_k[x, y] = \sum_{l=1}^{N_\lambda} H[x, y, l] \Phi_k[x, y, l].$$

If we vectorize $I_k[x, y]$ into a vector $i_k \in \mathbb{R}^{N_x N_y N_\lambda}$, we can express the measurement model as

$$i_k = X \Phi_k,$$

where $X \in \mathbb{R}^{N_x N_y N_\lambda \times N_k}$ is the matrix representation of the HSI. We can now formulate a convex optimization problem to solve for $i_k$:

$$\min_X \sum_{k=1}^{N_k} \|i_k - X \Phi_k\|^2 + R(X),$$

where $R(\cdot)$ is a regularizer. In our experiments, we use a combination of 2D total variation (TV) regularizer, and a 1D spectral smoothness regularizer, giving us

$$\min_X \sum_{k=1}^{N_k} \|i_k - X \Phi_k\|^2 + \eta_{TV} TV_{2D}(X) + \eta_{spectral} \|XD_k\|^2,$$

where $D_k$ is the 1D difference operator acting along spectral dimension. The values of $\eta_{TV}$ and $\eta_{spectral}$ are estimated empirically based on the number of captured images and provide a trade-off between robustness to noise and fidelity between measurements and reconstruction. We solved the optimization problem using stochastic gradient descent approach in PyTorch [Paszke et al. 2019]. We used Adam [Kingma and Ba 2014] optimizer for this purpose with a learning rate of $10^{-2}$ and $\beta_1 = 0.9, \beta_2 = 0.999$ for running averages of gradients and square of gradients, respectively. The optimization was run for a total of 200 iterations.

5.2 Reconstruction using Guide Image

The availability of an extra guide in the form of an RGB image can significantly increase the reconstruction accuracy. While it is possible to consider the RGB image as another linear measurement of the scene’s HSI and solve a joint optimization problem, we follow a non-linear approach that provides a better estimate. Our guided reconstruction technique is inspired by recent work in hyperspectral imaging [Saragadam et al. 2021], where superpixels are used to create a scene specific prior.

Given an RGB image $I_{RGB}[x, y]$ of the scene, we first partition the image into $Q$ superpixels using the simple linear iterative clustering (SLIC) [Achanta et al. 2012] algorithm. We make the assumption that spectral information is identical, up to scale, in this superpixel. With this model, we next solve the following regularized linear inverse problem at each superpixel using a simple least squares solution:

$$\min_{s_k} \sum_k \|i_{k, q} - g_{q}^T \Phi_{k, q}\|^2 + \eta \|s_k\|^2.$$

An overview of this approach is visualized in Fig. 12. The number of superpixels is dependent on the noise levels and number of images. As a general rule, the higher the noise, the fewer the superpixels.
so as to have a stable inversion. Similarly, the more the images, the smaller the superpixels.

6 SIMULATION RESULTS

To evaluate the proposed method, we simulate the acquisition setup on the ICVL [Arad and Ben-Shahar 2016] dataset which consists of several high spatial and spectral resolution hyperspectral images. Each hyperspectral image in the dataset has 519 bands in 400–1100nm range. To keep simulations close to our real setup, we pick hyperspectral images between 420nm and 940nm. We then compare our technique against existing snapshot techniques including Choi et al. [2017] which consists of a CASSI-type hardware and a deep neural network based reconstruction, and SASSI [Saragadam et al. 2021] which consists of a sparse spatio-spectral sampler along with RGB fusion. We downsample the spectra to 31 bands Choi et al. [2017], and 53 bands for all other approaches. We compare reconstruction with multiple images against LC cell architecture which involved capturing grayscale images modulated by a single LC cell. Our measurement model was similar to CS-MUSI [Oiknine et al. 2019], and the reconstruction was done with a least-squares approach. Unless otherwise specified, we assume a maximum light level of 1,000 electrons which, under photon noise and a read noise of 2 electrons, results in a signal to noise ratio of approximately 30 dB. For the proposed approach and LC cell, we use the linearized gamma measurement matrix described in Fig. 26(d). The constant \( \eta_{TV} \) in (12) is set according to estimated maximum light level \( \eta_{max} \) as \( \eta_{TV} = \frac{10^2}{\sqrt{\eta_{max}}} \), and the constant \( \eta_{spectral} \) is set to \( 5 \times 10^{-1} \). For reconstruction with guided image, the regularization constant \( \eta \) in (14) is chosen based on number of measurements with it varying linearly from \( 10^{-5} \) for a single image to \( 10^{-6} \) for 256 images.

We compute the accuracy of reconstruction using peak signal to noise ratio (PSNR) metric. We also compute spatially varying angular error as follows,

\[
\text{SAM}(x, y) = \cos^{-1}\left( \frac{\sum_{l=1}^{N_b} H(x, y, l) \tilde{H}(x, y, l)}{\sqrt{\sum_{l} H^2(x, y, l)} \sqrt{\sum_{l} \tilde{H}^2(x, y, l)}} \right),
\]

where \( H() \) is the ground truth HSI and \( \tilde{H}() \) is the estimated HSI. To enhance the result, we finally perform guided filtering of the resultant HSI with guide image [He et al. 2010] which resulted in up to 2 dB improvement in performance.

Comparisons against snapshot techniques. Figure 13 showcases reconstructed RGB image, spatial error map, and spectrum with various techniques using a single measurement with and without an extra guide image. We observe that the guide image significantly improves the performance with a 5 dB increase in PSNR and 5\(^{\circ} \) reduction in angular error. We also observe that reconstruction with guide image performs as good as other snapshot techniques with a small improvement over CASSI-type approaches [Choi et al. 2017].

Comparisons with multiple captures. Our primary competitor for multi frame reconstruction is a single LC cell capture [Oiknine et al. 2019] where images are captured with a spatially invariant spectral modulation. Figure 28 plots reconstruction accuracy with LC cell and our technique for varying number of images. Evidently, our approach with guided filtering produces superior reconstruction than LC cell for all number of measurements. We do observe that the reconstruction in the absence of a guide image is better than LC cell at fewer measurements and similar with 50 or more images. For a small number of images, it is more advantageous to spatially multiplex the various spectral filters. However with a large number of images, spatial multiplexing has a similar effect to capturing images with spatially invariant spectral filters.

The Appendix provides additional results in the form of performance across the 92 different patterns as well as light levels in the measurements (which is equivalent to differing measurement noise levels).
we use the full scan data to mimic the measurement process, and where the auxiliary RGB guide image is used. So we discuss the local diversity of patterns does play an important role in reconstruction performance. Looking closer, we can see that the gap between simulated and restored is much smaller for the 1D patterns; this can be attributed to the higher restoration performance for such patterns (as seen in Figure 10) as well as the lack of richness in the pattern becoming the bottleneck even for the simulated measurements. In contrast, 2D patterns have both a larger gap between the simulated and restored measurements as rendered RGB images. Below each RGB reconstruction, we provide the angular error against the full scan reconstructions; for these error maps, the brightest values are errors that are 20° or higher.

Fig. 16. Comparison of simulated, measured, and restored measurements. (a) Shown is an RGB image of the scene with inset of the spectral filter array that we used for the results in the other columns. (b-d) We visualize reconstructed HSIs using the simulated, measured, and restored measurements as rendered RGB images. Below each RGB reconstruction, we provide the angular error against the full scan reconstructions; for these error maps, the brightest values are errors that are 20° or higher.

For multi-image reconstructions, we greedily select a sequence of patterns that provide maximal diversification of filters. Specifically, starting with the best performing pattern from Figure 15, which is a 2D horizontal/periodic pattern, we sequentially add patterns that maximizes the addition of new spectral pixel over all previously selected patterns. The sixteen such patterns that we select with this scheme is visualized in Figure 17. We compare the reconstructions with rank-1 guided filter and guide-free TV prior in Figures 18 and 19. For the rank-1 approach, we linearly increase the number of superpixels used with the number of measurements. We also compare to reconstructions that would be obtained with just an LC cell, as opposed to an SLM, which would only provide global spectral modulation. As is to be expected, the proposed approach works significantly better than what we get with an LC cell, when we only have a few images. This is not surprisingly given the large ambiguities associated with the global spectral filters, that are only alleviated with additional measurements. The last row of Figure 19 shows a failure case of our approach; this is on a scene with an LED lamp that is both spatially and spectrally sparse, which makes it hard to recover with a single image. Multi-image techniques

Fig. 17. Patterns used for multi-image hyperspectral imaging. We greedily chose up to 16 patterns that ensured diversity of spectral filters at each pixel.

7 REAL RESULTS

Setup. We recover HSIs at a spatial resolution of 1024×1024 and a spectral resolution of 53 bands in the span of 420 to 940 nm. Since the spectral filters are linear in \( \frac{1}{\lambda} \), we sample these 53 bands uniformly in the reciprocal of the wavelength, which we observed provided a small increase in performance as opposed to the sampling linearly in the wavelength.

HSI reconstruction from a single spectrally-coded image. We first test the performance of reconstruction from a single spectrally-coded image. We observed that the performance of TV-based reconstructions were significantly worse than rank-1 reconstructions, where the auxiliary RGB guide image is used. So we discuss the results for just the rank-1 reconstruction here. Figure 15 provides the reconstruction performance of the guided rank-1 reconstruction for each of the 92 patterns described in Table 1 and Figure 10. We characterize performance with PSNR, measured in dB, and spectral angular error, measured in degrees. The spectral angular error is computed as the median angular mismatch between the estimated and ground truth spectrum observed at a pixel, as defined in (15). For the ground truth, we use the full scan data, acquired under 256 constant-valued SLM patterns; we use a least squares solver to recover the spectrum at each pixel individually. All plots provide aggregate results over the test scenes in Figure 7.

For each of the 92 patterns, we evaluate the performance using the raw measurements from the camera, the simulated data where we use the full scan data to mimic the measurement process, and the restored measurements use the deep neural network described in Section 4.4. The restored measurements generally outperform the raw measurements, often by a very large margin; this is fully consistent with our earlier observation in Figure 10, that the restoration network do reduce the mismatch to ground truth. Figure 16 provides an example of the reconstructions obtained with the simulated, measured, and restored measurements. Next, we observe that 2D patterns generally lead to better reconstructions in real experiments too, for the simulated as well as the restored measurements. This indicates that the local diversity of patterns does play an important role in reconstruction performance. Looking closer, we can

\[ \text{PSNR (dB)} \]

\[ \text{Angular Error (deg)} \]

\[ \text{Pattern Number} \]

\[ \text{Pattern Number} \]
perform better here, but both the guided and guide-free methods are ill-suited for such scenes.

**Color checkerboard.** Figure 20 visualizes the performance of the proposed technique on a color checkerboard illuminated with an incandescent lamp. We show reconstructions using the rank-1 guided filter for a single measurement, with the first pattern in Figure 17, as well as with 16 measurements. We compare the spectrum performance against the full scan reconstruction and observe a high level of match in the spectrum between the methods, with multi-image reconstruction performing consistently better.

8 DISCUSSIONS

This paper introduces a novel technique for spectral modulation — namely, a programmable and spatially-varying spectral filter array — and discusses its use in single- and multi-shot hyperspectral imaging. We achieve this capability using an LC-based phase SLM, and develop an optical schematic for implementing it while computationally handling unmodeled aberrations in the setup.

**Material classification.** A powerful capability enabled by our system is that of disambiguating between different materials in a scene from a single measurement using a carefully designed tiling of filters. An illustrative example of this is presented in Figure 21, where we seek to distinguish between a real plant, a fake one and the background. To do this, we first select three filters, out of the set of 256, such that the intensities associated with the materials are maximally separate (after projected to the 2D simplex to make them invariant to overall brightness); we find these three filters using a brute force search. We then tile these selected filters similar to a Bayer pattern, and capture a single image. Demosaicing this image allows us to cluster the normalized intensity at each pixel to one of the three classes, a result that agrees very well to obtain a “ground truth” result obtained by capturing three separate images, each with a distinct filter. This result indicates the immense potential of our system for applying adaptive sensing techniques on top of the setup.

**Spectral resolution.** The spectral resolution of our prototype is largely similar to what we expect with an LC cell-based device; specifically, we get a nonlinear spectral resolution with high resolution at smaller wavelength. We verify this with a scene illuminated with three lasers, and provide the reconstructions in Figure 22.

**Enhancing spectral diversity of filters.** The richness of the spectral modulation produced by our system relies critically on the range of phase retardation that can be implemented by the SLM. For our system, this range spans 3μm to 800nm — increasing this range is an important direction in enhancing the utility of our design. One way of realizing this is by introducing an LC cell in front of the SLM and using the additional phase retardation provided by it. Figure 23 visualizes the range of spectral filters we can obtain once we add such an LC retarder (Thorlab LCC1115-B) immediately in front of the SLM. The resulting setup implements spectral filters that have the form

\[
\frac{1}{2} - \frac{1}{2} \cos \left( \frac{2\pi n(x, y)}{d_{SLM}} + \frac{2\pi n_{LC}}{d_{LC}} \right),
\]

where the terms marked with “SLM” and “LC” denote to retardances applied at the SLM and LC cell, respectively. We observe that controlling the voltage across this retarder shifts the range of phase retardance that we can apply with our setup. Since the LC cell has the largest birefringence when no voltage is applied across it, we see the spectral filters with the largest oscillations at low voltages; however, note that the diversity of filters is low in this setting since the added phase retardance overwhelms the range of the SLM. At a slightly higher voltage (1.5V in Figure 23), we observe a different set of filters with greater diversity but fewer cycles. Hence, adding an LC cell opens up a novel and richer design space and is likely a powerful addendum to our design.

**Reconstruction algorithms.** Recent work on learning-based approaches have provided significant improvements in hyperspectral image reconstruction. Example of this include Choi et al. [2017] for the CASSI architecture and Gedalin et al. [2019] for LC cell-based imaging. These advances are largely complimentary to the optical setup introduced in this paper, i.e., we can expect the quality of reconstructions obtained with our setup to improve with the use of such sophisticated reconstruction algorithms.

**Miniaturization.** An important step toward practical adoption of our technique is the miniaturization of our setup. Our current setup involves 2 optical relays which increases the bulk as well as aberrations in the system. Further, it is important that we use lenses specifically designed to have little chromatic aberrations, which are extremely costly since we seek to image over a large spectral range. All of these concerns are ameliorated if we used a transmissive SLM instead of a reflective one, which would allow us to place the SLM directly on top of the sensor. This design would remove the need for both relays, since we can place the SLM and sensor at the image plane of the main lens. This implies that the issues with phase modulation and chromatic aberrations of the relay are all avoided. This design would likely become a viable alternative...
to many existing hyperspectral imaging techniques, once we have such transmissive SLMs with larger phase retardance and smaller pixel pitch.

ACKNOWLEDGEMENTS

This work was support in part by a Sony Faculty Innovation Award and the National Science Foundation under the following awards: 1652569, 1730147, and 1801382.

REFERENCES

Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. 2012. SLIC superpixels compared to state-of-the-art superpixel methods. IEEE Trans. Pattern Analysis and Machine Intelligence 34, 11 (2012), 2274–2282.

Boaz Arad and Ohad Ben-Shahar. 2016. Sparse Recovery of Hyperspectral Signal from Natural RGB Images. In European Conf. Computer Vision.

Yitzhak August and Adrian Stern. 2013. Compressive sensing spectrometry based on liquid crystal devices. Optics letters 38, 23 (2013), 4996–4999.

Seung-Hwan Baek, Incheol Kim, Diego Gutierrez, and Min H. Kim. 2017. Compact single-shot hyperspectral imaging using a prism. ACM Transactions on Graphics (TOG) 36, 6 (2017), 1–12.

Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. 2006. SURF: Speeded up robust features. In European Conf. Computer Vision. 404–417.

Bryce E Bayer. 1976. Color imaging array. US Patent 3,971,065.

Inchang Choi, Daniel S. Jeon, Giljoon 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.

Steven C. Gelbart, Red C. Thompson, and Anita Mahadevan-Jansen. 2007. Liquid-crystal tunable filter spectral imaging for brain tumor demarcation. Applied Optics 46, 10 (Apr 2007), 1896–1910.

Daniel Gedalin, Yaniv Oiknine, and Adrian Stern. 2019. DeepCubeNet: Reconstruction of spectrally compressive sensed hyperspectral images with deep neural networks. Optics express 27, 24 (2019), 35811–35822.

Michael Gehm, Remu John, Dave Brady, Rebecca Willett, and Timothy Schulz. 2007. Single-shot Compressive Spectral Imaging with a Dual-Dispenser Architecture. Optics Express 15, 21 (2007), 14013–14027.

Kaiming He, Jan Sun, and Xiaoou Tang. 2010. Guided image filtering. In European Conf. Computer Vision.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014).

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In ICLR.

Andy Lambrechts, Pilar Gonzalez, Bert Geelen, Philippe Soussan, Klaas Tack, and Murali Jayapala. 2014. A CMOS-compatible, integrated approach to hyper- and multispectral imaging. In IEEE International Electron Devices Meeting.

Wei Li, Changhui Jiang, Yuwei Chen, Juha Hyypää, Lingli Tang, Chunrong Li, and Shao Wei Wang. 2018. A liquid crystal tunable filter-based hyperspectral LiDAR system and its application on vegetation red edge detection. IEEE Geoscience and Remote Sensing Letters 16, 2 (2018), 291–295.
Fig. 20. Color checkerboard under an incandescent lamp. (a, b, c) Rendered RGB image for the full scan and restored measurements with single and 16 measurements. For the single image, we used the 2D horizontal/periodic pattern as the SLM pattern. (d) For each of the 4 × 6 squares, we plot the reconstructed SNR at the center of each square. The number on top of each plot indicates the spectral angular error between the estimated and ground truth spectra. The first number is for single image and the second for the 16 image reconstruction. We can observe a very high level of match to the ground truth. Given that the first row has the same spectrum under different light levels, we can get a sense of the noise performance of our approach (and of the full scan reconstructions).

Xing Lin, Gordon Wetzstein, Yehun Liu, and Qionghai Dai. 2014. Dual-coded Compressive Hyperspectral Imaging. *Optics Letters* 39, 7 (2014), 2044–2047.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *ECCV*.

Ankit Mohan, Ramesh Raskar, and Jack Tumblin. 2008. Agile Spectrum Imaging: Programmable Wavelength Modulation for Cameras and Projectors. In *Comp. Graphics Forum*.

Srinivasa G Narasimhan and Shree K Nayar. 2005. Enhancing resolution along multiple imaging dimensions using assorted pixels. *IEEE Trans. Pattern Analysis and Machine Intelligence* 27, 4 (2005), 518–530.

Yaniv Oikhnine, Isaac August, Vladimir Farber, Daniel Gedalin, and Adrian Stern. 2019. Compressive sensing hyperspectral imaging by spectral multiplexing with liquid crystal. *Journal of Imaging* 5, 1 (2019), 3.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*.

Fig. 21. Disambiguating between materials using programmable spectral filter arrays. (a) We image a scene with plants, real and plastic. (b) The measurement trace, as a function of SLM input index, is visualized for the two materials as well as the background. We find three index values, marked with dotted vertical lines, that lead to maximally different measurements for the three materials. (c) An image of the scene is captured with an SLM displaying a checkerboard pattern comprising of the two chosen index values. The inset is the zoomed in version of the cropped region marked in red. (d) Using this single measurement, we can now create a metric that maximally disambiguates between the two materials and threshold it in (e) to get a material map. For comparison, the label map from the full scan dataset is shown in (f).

Xing Lin, Gordon Wetzstein, Yehun Liu, and Qionghai Dai. 2014. Dual-coded Compressive Hyperspectral Imaging. *Optics Letters* 39, 7 (2014), 2044–2047.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *ECCV*.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI* 234–241.

Ashish Vigwani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, I. ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*.

Shin-Tson Wu, Uzi Efron, and LaVerne D. Hess. 1984. Birefringence measurements of liquid crystals. *Appl. Opt. 23, 21 (Nov 1984), 3911–3915.*

Fig. 22. Spectral resolution. To characterize the spectral resolution, we visualize the reconstructions, with full scan measurements, of three lasers. We obtain a FWHM bandwidths of 23.9nm at 532nm, 31.9nm at 635nm and 69.0nm at 850nm. As is consistent with LC cells, the device obtains a bandwidth that is roughly proportional to the wavelength.
We briefly go over the principle of operation of an LC cell when a sinusoidal voltage is applied across the LC cell. This filter is sinusoidal in the wavenumber, or the reciprocal of the wavelength. The frequency of this sinusoid is determined by the term \( \Delta \nu \), which quantizes the path difference introduced by the LC cell. As is to be expected, thicker LC cells introduce a larger path difference which creates spectral filters with more cycles over \( \lambda \). Similarly, higher levels of birefringence \( \Delta n \), that typically happens for low values of \( v \), also creates a larger number of cycles over the wavelength of interest.

### A Basics of Spectral Filtering with LC Cells

We briefly go over the principle of operation of an LC cell when used to implement a spectral filter. The reader is referred to [Wu et al. 1984] for a detailed treatment of this material.

The basic imaging setup, seen in Figure 24, consists of an LC cell that is sandwiched between two linear cross polarizers, with their polarization axes oriented at \( \pm 45^{\circ} \) to the LC cell’s fast axis. To describe the propagation of light through this stack, we use the Jones vector formulation, with the two components aligned with the fast and slow axes of the LC cell. Unpolarized light at a wavelength \( \lambda \) incident on this filter is first linearly polarized by the first polarizer; since the axis of polarization is at \( 45^{\circ} \) to the fast/slow axes of the LC cell — which we use to define the coordinate axes of a Jones vector, this light can be represented using the Jones vector

\[
\frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}.
\]

The birefringence of the LC cell introduces an optical path delay between its fast and slow axes that is equal to

\[
\Delta n(v) d_{LC},
\]

where \( d_{LC} \) is the thickness of the LC cell, \( v \) is the RMS voltage applied across it\(^2\), and \( \Delta n(v) \) is the resulting birefringence at this voltage. This results in a phase difference of

\[
\phi(\lambda) = 2\pi \frac{\Delta n(v) d_{LC}}{\lambda},
\]

and hence, Jones vector after the LC cell is given as

\[
\frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ e^{j\phi(\lambda)} \end{bmatrix}.
\]

\(^2\)Typically, an LC cell is operated by applying a high-frequency signal to avoid ion formation and damage to the device. The specifics of this often controlled by the driver associated with the device. So, when we refer to the control voltage, we refer to the RMS value as opposed to the specific waveform used.

The second polarizer changes the Jones vector to

\[
\frac{1}{2\sqrt{2}} \begin{bmatrix} 1 - e^{j\phi(\lambda)} \\ 1 \end{bmatrix},
\]

The intensity of the light that exits the filter is given as the square of the magnitude of the Jones vector, which evaluates to

\[
\frac{2}{8} |1 - e^{j\phi(\lambda)}|^2 = \frac{1}{4} (2 - 2\cos(\phi(\lambda))) = \frac{1}{2} \left(1 - \cos \left(2\pi \frac{\Delta n(v) d_{LC}}{\lambda}\right)\right)
\]

The expression in (21) provides spectral filter observed when an RMS voltage \( v \) is applied across the LC cell. This filter is sinusoidal in the wavenumber, or the reciprocal of the wavelength. The frequency of this sinusoid is determined by the term \( \Delta \nu \), which quantizes the path difference introduced by the LC cell. As is to be expected, thicker LC cells introduce a larger path difference which creates spectral filters with more cycles over the wavelength. Similarly, higher levels of birefringence \( \Delta n \), that typically happens for low values of \( v \), also creates a larger number of cycles over the bandwidth of interest.

### B Calibration

The proposed system requires key calibration steps that need to be performed for correct operation. First, we need to measure the spectral filter created by the LC cells under different input voltage, or what we refer to as spectral calibration. And second, we need to get a pixel-to-pixel mapping between the SLM and the camera(s); this step is especially important since our object is to implement a filter array and hence, this XY calibration provides critical information on the filter applied to obtain the measurement at a sensor pixel.

**Spectral calibration.** We perform spectral calibration by mounting a spectrometer (OceanOptics FLAME-S-VIS-NIR-ES) in place of the main camera and sweep 256 constant-valued patterns on the SLM, corresponding to its 8-bit input. For each displayed pattern, we capture a spectral measurement over the range [380, 1010]nm at a spectral resolution of 0.35nm. Recall, that the actual voltage across the LC cells in the SLM is determined by the gamma curve.

**SLM-to-camera alignment.** To recover hyperspectral images from the measurements acquired with the main camera, we need the knowledge of which SLM pixel is observed at each camera. With...
ideal optics, this mapping would be given by a homography. However, due to the presence of radial distortions in the image, we needed to use a more complex model for the fit.

To get the SLM to sensor mapping, we captured a collection of images, with the device imaging a diffuser that was uniformly lit with a broadband light source. Each captured image was of the SLM displaying an all-zero pattern, except on a single row or column that was set to 255. Figure 25(a) provides an example of such an image, where the SLM displays a single row. We detect the location of the line on the sensor coordinates using simple comparison against a reference capture with all-zero pattern on the SLM. After some pre-processing, we obtain a collection of pixels on the sensor that maps to this particular row on the SLM. The SLM sweeps through multiple rows, staggered by a small amount — 33 pixels in our implementation. At the end of this row scan, we end up with a collection of \( N \) correspondences of the form \( \{(x^s_i, y^s_i)\}_{i=1}^N \) where the sensor locations \((x^s_i, y^s_i)\) map to SLM row \( y^s_i \). We fit a third-order polynomial that maps \((x, y^s)\) to \( y^s\), using RANSAC to obtain a robust fit. The same process is repeated for columns of the SLM and, as with the rows, we obtain a third-order polynomial fit.

Once we have the polynomial mapping from the sensor to the SLM, we invert this mapping by generating a large number of point-to-point correspondences, and applying the polynomial fit for the rows and columns in reverse — thereby obtaining two cubic polynomials that relate \((x, y^s)\), a point on the SLM, to \( x^s \) or \( y^s \), the corresponding row/column on the sensor. We visualize the accuracy of this fit in Figure 25(b), where we display a grid pattern on the SLM — with a spacing that is different from what was used previously in row/column sweep — and overlay the predicted grid location on top of the sensor image. We generally observe an accurate match between the measured and observed grid, even at the very boundaries of the captured image. Quantitatively the average error in calibration is approximately 0.1 pixels, with a maximum error of less than 0.5 pixels.

We also use the grid image shown in Figure 25(b) to incrementally correct a pre-calibrated system using just a single capture, along with an all-zero reference for comparison. Such a correction, applied periodically, allows us to account for small displacements due to mechanical vibration.

**Guide alignment.** Aligning the guide RGB camera to the main camera was done by imaging a highly-textured scene with the SLM displaying a constant pattern, and hence, no spatially-varying filters. We then used standard image registration techniques by detecting SURF feature points and descriptors [Bay et al. 2006], establishing correspondences, and fitting a homography model to it.

### C DESIGNING THE GAMMA CURVE

The construction of the gamma curve requires knowledge of the term \( d_{LC} \frac{\partial n}{\partial V} \). This depends on the term \( d_{LC} \Delta n(\cdot) \), which we estimate using the following procedure.

**Step 1 — Spectral filters as a function of input voltage.** We place a spectrometer in place of the sensor in Figure 24, and sweep a constant pattern at known voltage on the SLM. For the SLM that we used, which we describe in more detail later, the working range with best diversity of spectral filters was \([0, 4.2]V\), and hence, we linearly scanned the SLM in this range and measured the resulting spectral filter for each voltage. The resulting collection of spectral filters is visualized in Figure 26(a).

**Step 2 — Estimating \( \Delta n(v)d_{LC} \) as a function of input voltage \( v \).** In an ideal setting, the spectral filter observed would be a perfect fit to (21) and hence, we can estimate the value of \( \Delta n(v)d_{LC} \) for each input voltage \( v \) analytically. However, in practice, non-idealities in the SLM as well as optics used produces deviations in the spectral filter. So, we perform a brute force search over a range of values \( v \) — in this case, \([300, 3000]nm\) — and finding the value that has the least misfit to the analytical expression given in (21). This procedure provides us with an empirical estimate of \( \Delta n(v)d_{LC} \), as shown in Figure 26(b).

**Step 3 — Estimating \( g \).** Our goal is to estimate a function \( g \) such that \( d_{LC} \Delta n(\cdot)(V) \) is linear in its argument. Recalling that our SLM is controlled via a 8-bit video signal, there are 256 possible inputs to the \( g \) function which maps to a voltage value that is applied to the SLM. We also need to ensure that this gamma function preserves the range of voltages such that \( g(0) = 0V \) and \( g(255) = 4.2V \). To achieve this we sample 256 uniformly-placed values between the \( \Delta n(0V)d_{LC} \) and \( \Delta n(4.2V)d_{LC} \), the maximum and minimum values taken. For each value, we find the voltage \( v \) that achieves the specified birefringence. This value of voltage is what the \( g \) function maps onto, for each of its 256 inputs. This resulting \( g \) for our SLM is visualized in Figure 26(c).

As noted earlier, this choice of \( g \) makes the phase gradient solely dependent on the spatial gradients of the pattern \( p(x, y) \) that we display on the SLM.

### D NETWORK DESIGN AND TRAINING

The number of channels when passing through the downsampling blocks are 192, 384, 768, and 768, and the upsampling blocks have them in reverse. Each downsampling block comprises of a convolution layer of kernel size \( 3 \times 3 \) followed by a convolution layer of kernel size \( 4 \times 4 \) with two strides for the downsampling operation. Each upsampling block comprises of a convolution-transpose layer...
Programmable Spectral Filter Arrays for Hyperspectral Imaging

Fig. 26. Procedure for designing the SLM gamma curve. Our goal is to estimate a gamma function such that the nonlinear relationship between SLM birefringence and input voltage is linearized. To do this, we use a spectrometer to measure the spectral filter produced over a range of input voltages. The measured data is shown in (a). For each voltage $v$, we brute-force search for the value of $\Delta n(v)$ that fits the spectral filter obtained at that voltage by measuring accuracy to (21). The resulting loss function is visualized as an image in (b). Identifying the minimum for each voltage provides us with value of $\Delta n(v)$, overlaid as a red curve in (b). We use this estimated value to design a gamma curve $\gamma(\cdot)$ for the SLM, shown in (c), such that the overall function $\Delta n(\gamma(\cdot))$ is linear. (d, e) The resulting set of filters that we obtain, now indexed as a function of the 8-bit index used in controlling the SLM.

Training. We use the measurement and simulated images of 15 scenes, each with multiple SLM patterns, as our training set. We randomly pick $64 \times 64$ patches from measurement and simulation images as training input-output pairs. Similarly, we use patches from measurement and simulated images of eight other scenes for validation. We use L2-norm of the difference between the estimated and ground-truth images as our loss function to be minimized. We avoid image priors such as gradient sparsity since our operative images are not “natural.” We use Adam optimizer [Kingma and Ba 2015] with a learning rate of $10^{-3}$, $\beta_1 = 0.9$, and $\beta_2 = 0.999$. We train the network for 100k iterations with a batch size of 500 that takes around 45 hours using four Titan Xp GPUs.

E ADDITIONAL SIMULATION RESULTS

Comparisons against snapshot techniques. We simulate reconstruction with all 92 patterns to understand their effect on accuracy. Figure 27 shows a plot of average PSNR for six test HSIs for various pattern types with rank-1 and TV reconstruction. Since 2D patterns have diverse spectral filters in a local neighborhood, the reconstruction is expected to be more accurate than other patterns, as verified by the plot.

Comparisons with multiple captures. Our primary competitor for multi frame reconstruction is CS-MUSI [Oiknine et al. 2019] where images are captured with a spatially invariant spectral modulation. Figure 28 plots reconstruction accuracy for CS-MUSI and our technique for varying number of images. Evidently, our approach with guided filtering produces superior reconstruction than CS-MUSI for all number of measurements. We do observe that the reconstruction in the absence of a guide image is better than CS-MUSI at fewer measurements and similar with 50 or more images. For a small number of images, it is more advantageous to spatially multiplex the various spectral filters. However with a large number of images, spatial multiplexing has a similar effect to capturing images with spatially invariant spectral filters. We also compared CS-MUSI against our technique with varying light levels in Fig. 29. We noticed that our approach is more robust to noise, especially at very low to low light levels – this advantage primarily arises from the use of guide image which acts as a regularizer for the spatial dimension.
Fig. 28. **Multi-frame performance.** Our proposed approach performs better than CS-MUSI with small number of images or lower light levels. Performance increases uniformly with increasing number of images and light levels, and outperforms CS-MUSI on an average. With 256 images, performance is same as CS-MUSI – this is to be expected as the measurements with proposed approach and with LC Cell are equivalent.

Fig. 29. **Performance with varying light levels.** Experiments were performed with 16 images for all approaches. The plots show PSNR and angular error as a function of light level, while the bottom images show angular error map for LC cell-based reconstruction and our rank-1 reconstruction. At very low to medium light levels, rank-1 outperforms LC cell-based measurements by a large margin. When the light levels are very high, LC cell has a better accuracy, which can be attributed to accurate spatial reconstruction.