

Programmable Spectrometry: Per-pixel Material Classification using Learned Spectral Filters

Vishwanath Saragadam, *Member, IEEE*, and Aswin C. Sankaranarayanan, *Senior Member, IEEE*

Abstract—Many materials have distinct spectral profiles, which facilitates estimation of the material composition of a scene by processing its hyperspectral image (HSI). However, this process is inherently wasteful since high-dimensional HSIs are expensive to acquire and only a set of linear projections of the HSI contribute to the classification task. This paper proposes the concept of programmable spectrometry for per-pixel material classification, where instead of sensing the HSI of the scene and then processing it, we optically compute the spectrally-filtered images. This is achieved using a computational camera with a programmable spectral response. Our approach 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 ample training data, we use learning techniques to identify the bank of spectral profiles that facilitate material classification. We verify the method in simulations, as well as validate our findings using a lab prototype of the camera.

Index Terms—Computational Photography, Hyperspectral Image, Programmable Filter, Material Identification

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 [7], tagging of key cellular components in fluorescence microscopy [23], land coverage and weather monitoring [6], [15], and even the study of chemical composition of stars and astronomical objects using line spectroscopy [17].

While spectral profiles and its imaging variant, hyperspectral images (HSI), have found application in computer vision tasks [19], [34], [46], widespread adoption has been hindered due to inherent challenges in its 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. 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.

This paper 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 $\mathcal{O}(K)$ 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

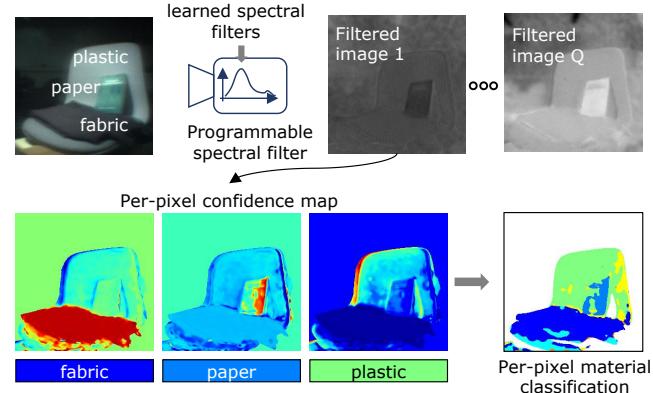


Fig. 1: **Spectrum-based material classifier.** We propose an optical setup that is capable of classifying material on a per-pixel basis. This is achieved by optically computing a small set of spectrally-filtered images of the scene and then performing classification.

translates to optically filtering the scene's HSI using known spectral filters, which can be achieved if we had a spectrally-programmable camera.

We propose an 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; this process is illustrated in Fig. 1.

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

- 1) *Optical computing for spectral classification.* By optically computing the linear projections of spectral profile at each pixel, we circumvent the need for sampling of the full HSI. This requires significantly fewer measurements and has higher

• The authors are with the ECE department, Carnegie Mellon University, PA, 15213. E-mail: saswin@andrew.cmu.edu

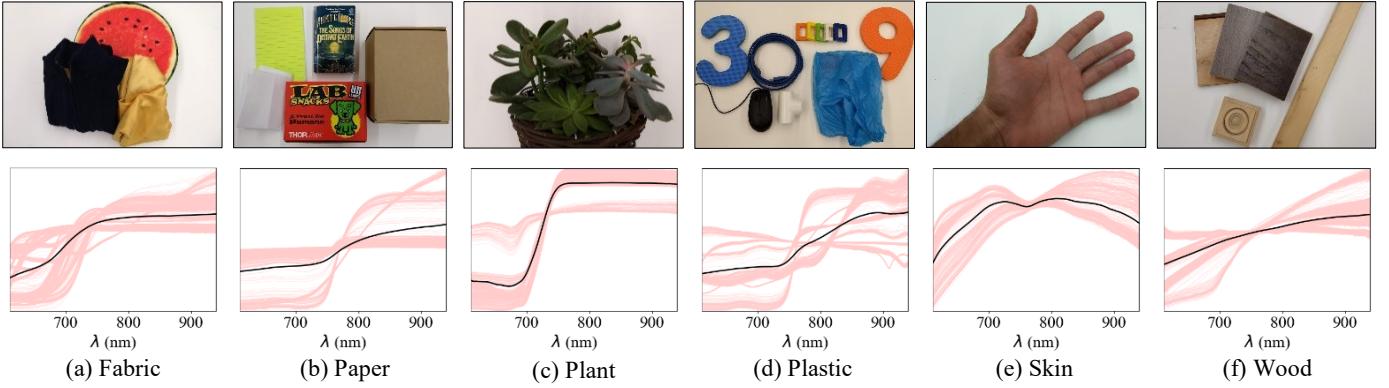


Fig. 2: **Spectrum of materials.** Spectrum is often a unique identifier of materials and can be used for per-pixel classification tasks. We collected spectral profiles of several everyday materials and have shown images of some of the objects we sampled as well as the spectral profiles, with the black 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.

light efficiency due to broadband spectral filters.

- 2) *New spectrometric dataset.* Since material classification only requires spectral profiles, we show that discriminant filters can be learned on labeled spectral measurements. To achieve this, we collected several spectral profiles of everyday objects, and learned discriminant filters (see Fig. 2). The codebase for learning the filters, and our new dataset can be downloaded from [1].
- 3) *Lab prototype.* 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.

2 PRIOR WORK

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

2.1 Spectral classification

Given spectral profile $s(\lambda)$, often obtained using a spectrometer, the goal of spectral classification is to estimate the material composition of the object. Given $s(\lambda)$, we can classify the material using many different techniques including support vector machines (SVM) and neural networks [10]. This process is often computationally light weight, and accurate — as spectrum is a unique identifier of materials. However, its extension to classification of all pixels in the image of scene is a challenging task. Classification at image level requires capturing spectrum at each pixel — thereby requiring a full scan of the hyperspectral image which is often daunting.

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,

$$H(x, y, \lambda) = \alpha(x, y) S_{L(x, y)}(\lambda), \quad (1)$$

where $L(x, y)$ is label of the material contributing to spectrum at (x, y) , and $\alpha(x, y)$ is a scaling parameter. Note that the model above assumes all spatial pixels are pure, i.e., every pixel gets contribution from only one material.

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 [9]. Specifically, the HSI of the scene is decomposed as,

$$H(x, y, \lambda) = \sum_{k=1}^K s_k(\lambda) a_k(x, y), \quad (2)$$

where $s_k(\lambda)$ is the spectra of k^{th} material, and $a_k(x, y)$ 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, $\{(d_k(\lambda), \beta_k)\}_{k=1}^M$ that generate per-pixel feature vector via spectral-domain filtering:

$$F_k(x, y) = \int_{\lambda} H(x, y, \lambda) d_k(\lambda) d\lambda + \beta_k. \quad (3)$$

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 the *de facto* way of HSI classification [11], [28] in the post-processing step. More sophisticated learning techniques based on neural networks use spectral features [18] or spatio-spectral features [5], [14], [16], [20], [22], [24], [27], [45] for classification.

2.3 Need for high spectral resolution

A key requirement for accurate classification is spectral measurements at high resolutions [8]. Sampling at resolutions 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 verify this, we performed a 6-class classification with

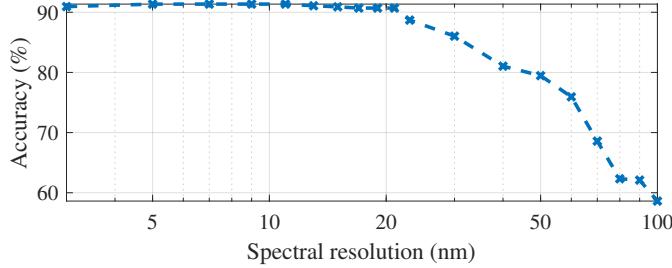


Fig. 3: **Effect of spectral resolution on accuracy of material classification.** In order to classify accurately, it is important to sample the spectral profiles at high resolutions. The plot shows classification as a function of spectral resolution for a 6-class classification task with a one-vs-rest SVM classifier. A high spectral resolution ensures a high accuracy in material classification; however, such high resolutions present challenges in acquisition that we mitigate by performing the necessary computations in the optical domain.

spectrometric data measured at varying resolutions. A glimpse of the data is presented in Fig. 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 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 approaches which require fewer measurements than HSI dimensionality, without sacrificing classification accuracy.

2.4 Compressive inference for spectral classification

A class of techniques, based on compressive sensing (CS) [2], seeks to reduce the measurements required by capturing random linear projections of the signal of interest. Since random projections are known to preserve the geometry of certain signal domains, we can perform classification on the low-dimensional projections. In the context of spectral classification, Li et al. [21] perform this by capturing a small set of spatially-multiplexed measurements and then recovering labels via an unmixing framework. Ramirez et al. [38], [39], [40] 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 computations to the optical domain.

2.5 Optical computing

Our approach relies on the idea of *optical computing*, in that, we seek to perform the linear operations required for classification directly in the optical domain and circumvent the need to fully scan the HSI. Optical computing has found use in various vision tasks such as low-rank approximation of light transport matrices [33] and hyperspectral images [44], as well as spectral classification using programmable light sources [12], [35]. Our approach

is similar, in spirit, to methods that per-pixel classification from BRDFs by varying the incident illumination [13], [25], or using first layer of a neural network to capture light fields [4]. To reduce the complexity of the hardware required, we consider a simpler approach which relies only on the spectral profiles for classification so as to process each pixel individually. Such a strategy is less accurate than spatial and spectral versions [5], [14], [16], [20], [22], [24], [27], [45], but significantly reduces the complexity of the imaging system.

2.6 Dynamic spectral filters

To implement our spectral classification technique, we require a camera that is capable of capturing images with arbitrary spectral filters. Spectral filtering can be achieved by modifying illumination spectra [41], or the spectral 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 techniques to achieve spectral filtering have been proposed in the past. Agile spectral imager [29] rely on the coding the so-called “rainbow plane” to achieve arbitrary spectral filtering. This was further developed by [26] 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 pupil codes such as open aperture or a slit directly tradeoff spatial resolution for spectral resolution. This was identified in [26], and [43] 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 accurate spectral classification.

We instead rely on the optical setup in [44] to overcome this 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 [44] is designed for HSI image acquisition; this paper adapts the underlying ideas for performing per-pixel material classification.

3 PROGRAMMABLE SPECTRAL FILTER

Our optical setup is a modification of the optical setup proposed in [44] and is shown in Fig. 4(a). The setup consists of an objective lens focusing scene’s image on a diffraction grating on plane P1, and a series of lenses that relay the image on P1 to image sensor on P3. The intensity formed on plane P2 represents the spectral content of the scene and is called the rainbow plane. The length of the relay system, from diffraction grating to image sensor is four focal lengths, and hence is called a $4f$ system. Next, we explain the relevant parts of the signal propagation.

3.1 4f system for spectral programming

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,

$$I_4(x, y) = a^2(-x, -y) * \left(S\left(\frac{x}{f\nu_0}\right) \tilde{c}\left(\frac{x}{f\nu_0}\right) \right), \quad (4)$$

where $S(\lambda) = \int_{(x,y)} H(x, y, \lambda)$ is spectrum of the scene, $\tilde{c}(\lambda)$ is response of the optical system, ν_0 is the density of grooves

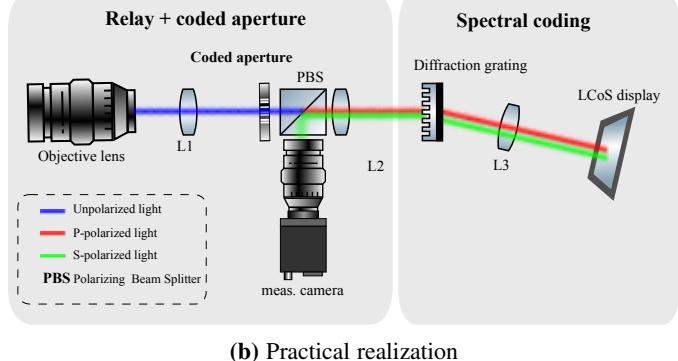
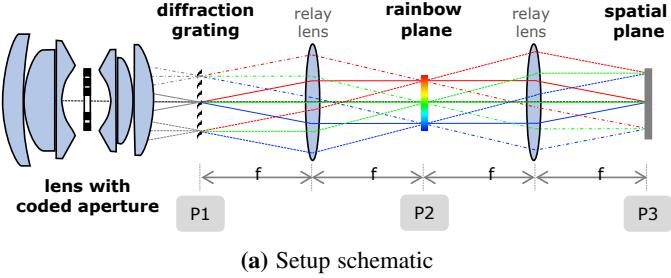


Fig. 4: **Schematic for programmable spectral filter.** The optical architecture in (a) consists of a lens assembly with coded aperture which introduces spatial and spectral blurs. By placing an SLM in P2, the HSI of the scene can be spectrally filtered and sensed by a camera sensor on P3. (b) shows a compact realization of the optical setup.

in mm^{-1} , and $*$ represents linear convolution. The intensity on image plane P3,

$$I_5(x, y) = \int_{\lambda} \left(H(x, y, \lambda) * \left| \frac{1}{\lambda^2 f^2} A \left(-\frac{x}{\lambda f}, -\frac{y}{\lambda f} \right) \right|^2 \right) d\lambda, \quad (5)$$

where $A(u, v)$ is the 2D Fourier transform of $a(x, y)$. The key observation from (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 shown in Fig. 5, 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 [44], as it is designed to promote invertibility in both domains.

3.2 Optical setup

Our optical setup is in principle similar to Fig. 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 [44] is placed in the lens assembly. Figure 4(b) shows a schematic of a practical implementation of the same optical setup. Lenses L1, L2 relay the image plane on to the diffraction grating, with a coded aperture between the two lenses. The diffraction grating splits the input light into constituent wavelengths, which is then focused by lens L3 on the LCOS SLM. The SLM reflects back light in the same direction, and L3 focuses image back on to the diffraction grating, which recombines wavelengths. Finally, L2 and the NIR camera relay back the image on the diffraction grating. In effect, we achieve a chromatic aberration-free spectral coding with the proposed optical setup.

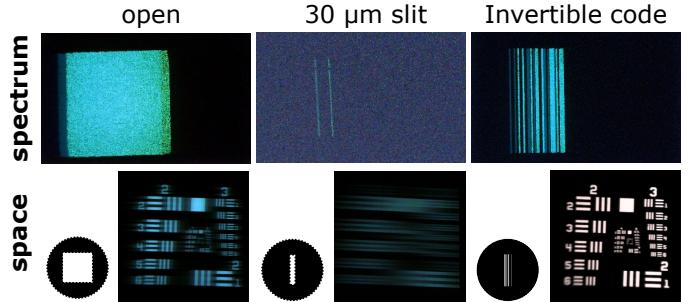


Fig. 5: **Spatio-spectral resolution tradeoff.** A slit is capable of high spectral resolution whereas an open aperture is capable of high spatial resolution but both are inappropriate for high spatio-spectral HSI imaging. In contrast, a coded aperture introduces an invertible spatial and spectral blurs which can then be deconvolved. Figure reproduced with permission from [44].

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,

$$\hat{i}(x_3, \lambda) = \int_{x_2} p_k(x_2) a(-x_2 + \lambda f \nu_0) i(x_1, \lambda) dx_2. \quad (6)$$

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

$$\begin{aligned} \int_{\lambda} i(x_1, \lambda) s_k(\lambda) d\lambda &= \int_{\lambda} \hat{i}(x_3, \lambda) d\lambda \\ &= \int_{\lambda} \int_{x_2} p_k(x_2) a(-x_2 + \lambda f \nu_0) i(x_1, \lambda) dx_2 d\lambda \\ \implies \int_{x_2} p_k(x_2) a(-x_2 + \lambda f \nu_0) dx_2 &= s_k(\lambda), \end{aligned} \quad (7)$$

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. (7), and is given by,

$$p_k(x_2) = \text{deconv} \left(s_k \left(\frac{x_2}{f \nu_0} \right), a(-x_2) \right), \quad (8)$$

where ‘‘deconv’’ is a simple 1D deconvolution function, such as Wiener deconvolution. Figure 6 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 LEARNING DISCRIMINANT FILTERS

Our camera is capable of implementing any classifier that relies on a linear projection as a first step. We first explain binary classification with support vector machines and then multi-class classification with neural networks.

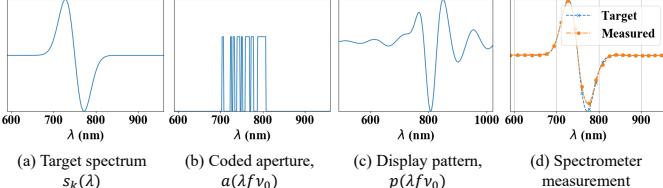


Fig. 6: **Effect of coded aperture.** To display a desired spectral filter (a) $s_k(\lambda)$, we need to project a pattern such that the convolution of the (c) projected pattern with the flipped coded aperture is equal to the target filter. (d) compares the output of our optical setup with a spectrometer and the target spectral filter.

4.1 Support vector machine

SVMs provide a binary, linear classifier by learning a separating hyperplane on the training dataset. Given spectral measurements $s(\lambda)$, classification with SVM involves the following operation,

$$\int s(\lambda)w(\lambda)d\lambda + c_0 \begin{cases} \leqslant 0, & \text{class 1} \\ > 0, & \text{class 2} \end{cases} \quad (9)$$

where $w(\lambda)$ is the learned separating hyperplane and c_0 is the bias term. Noting the similarities between (3) and the equation above, we see that binary classification requires capturing a single image with $w(\lambda)$ as the spectral filter. Therefore, we can perform per-pixel, binary material classification with just a single measurement, instead of scanning the full spectrum $s(\lambda)$.

4.2 Deep neural networks

Deep neural networks (DNNs) provide a richer alternative to SVMs. The architecture used in this paper is illustrated in Fig. 7. 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, 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 classes. 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 [3] and processing light fields [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 8 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.

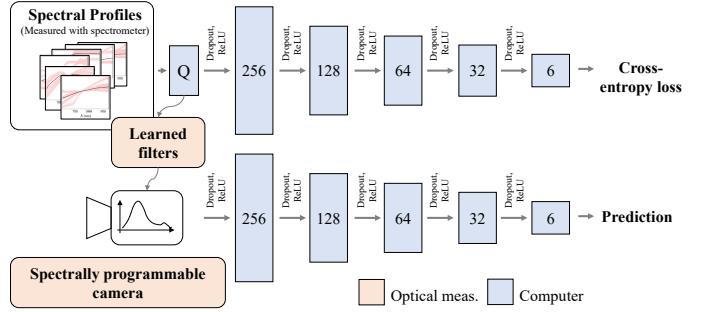


Fig. 7: **Proposed optical classifier.** The proposed optical classifier broadly consists of two stages. In the first stage, we learn the weights of a neural network with spectrum as input and class label as output. The training process outputs the set of discerning filters, marked “learned filters” 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.

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)
This paper	DNN	Linear, spectrum-only	16	90% (Computed)

Fig. 8: **Simulations on the Indian Pines dataset.** We compare state-of-the-art classifiers against the classifiers proposed in this paper. 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.

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.

5.1 Lab prototype

Figure 9 shows a photograph of the lab prototype we built along with labels for relevant components. A detailed optical layout along with the list of components is in the supplementary material. 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 (Hamatsu ORCA Flash 4.0 LT). Inspired by previous work in material classification [10], [42], 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 10 shows an example of captured HSI of an acrylic painting.

Calibration. Our optical setup broadly requires calibration of the code resulting in spectral blur, calibration of wavelengths and

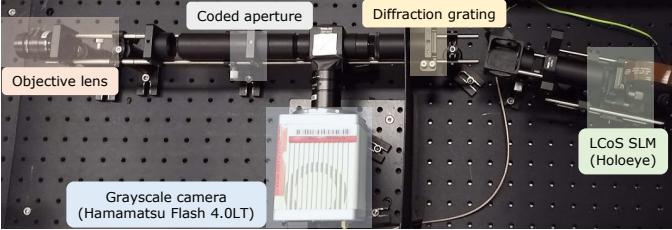


Fig. 9: **Lab prototype.** The picture shows the lab prototype we built with only the major components marked. Please refer to supplementary for complete details. We used an objective lens of 50mm focal length, while all other lenses were 100mm.

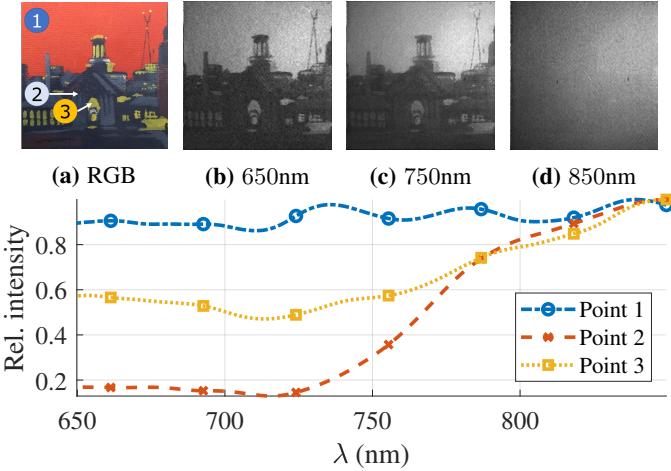


Fig. 10: **Example HSI.** Our prototype is designed to capture images from 600nm to 900nm. (a) was captured using a cellphone while (b)-(d) are images captured by our setup. Bottom row shows spectral profiles at three marked points.

finally, spatial PSF. We use narrow-band lasers for calibrating both code and wavelengths, and use a $10\mu\text{m}$ pinhole for calibrating spatial PSF. Details are available in the supplementary material.

5.2 Handling illumination spectrum

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 $\hat{H}(x, y, \lambda) = H(x, y, \lambda)c(\lambda)l(\lambda)$, 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 $\hat{s}_k(\lambda) = s_k(\lambda)/(c(\lambda)l(\lambda))$, 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 $\hat{H}(x, y, \lambda)$, 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.

5.3 Dataset

To learn discriminant spectral filters, we collected spectra of several everyday object with a OceanOptics Flame FX 400 – 1000nm

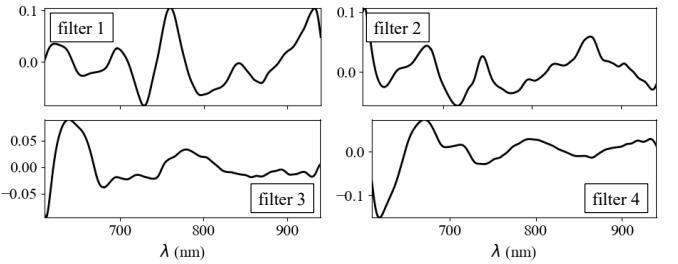


Fig. 11: **Learned filters.** The filters shown here correspond to a DNN classifier with 4 spectral filters.

spectrometer. For experiments in this paper, 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. 2. More details can be found in supplementary material. The code to learn the spectral filters, and the spectrometer dataset can be downloaded from [1].

5.4 Training classifiers

For binary classification, we used SVM to learn the spectral filters. We used Scikit-Learn [37] for this purpose. We trained DNNs for multi-class classification with the network architecture shown in Fig. 7 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 [36] 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 filters. The learned filters are shown in Fig. 11. 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.

5.5 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,

$$I_{\text{sum}}(x_0, y_0) = \int_{\lambda} H(x_0, y_0, \lambda) d\lambda \quad (10)$$

Then, the measurements for classification are,

$$\tilde{I}_k(x_0, y_0) = \frac{I_k(x_0, y_0)}{I_{\text{sum}}(x_0, y_0)}, \quad (11)$$

where $I_k(x_0, y_0)$ is the measurement with spectral filter $s_k(\lambda)$. Such an approach also makes the measurements invariant to shading of individual pixels [30], [31], [32]. We scale the spectra the same way even while training, which makes the scaling consistent.

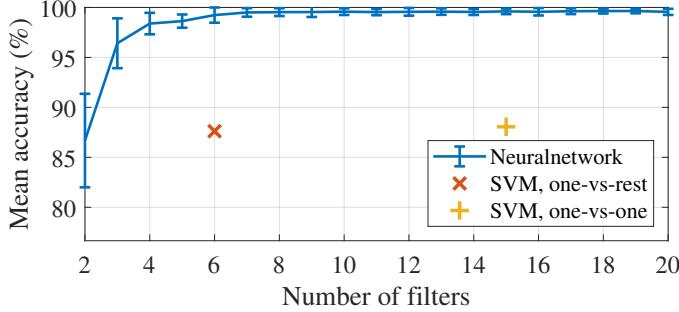


Fig. 12: **Accuracy vs. number of filters.** We varied the number of filters and estimated mean accuracy on a held out test dataset. Based on the plot, we observe that DNNs outperform SVM, and that accuracy starts saturating beyond 4 filters.

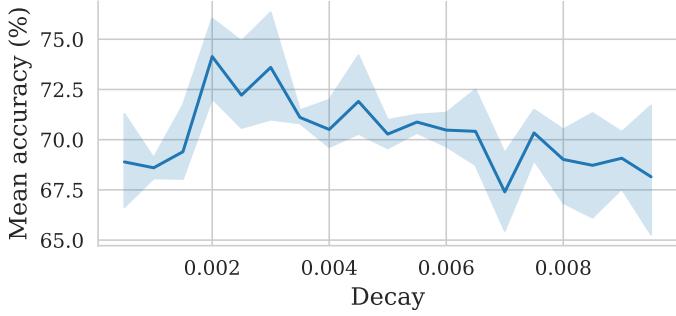


Fig. 13: **Accuracy vs. smoothness.** We plotted accuracy as a function of decay constant, η , with lower η encouraging smoother filters. The accuracy is highest around $\eta = 2 \times 10^{-3}$, which was the value we used for all our experiments.

Hence any set of measurements with spectral profiles requires just one extra image.

5.6 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. 12 shows an accuracy plot. We observe that DNNs achieve higher accuracy than SVM. Further, for our 6-class problem, we see that the accuracy saturates after four filters. We hence chose $Q = 4$ in all our experiments.

5.7 Accuracy vs. smoothness

Smoothness of spectral filters is controlled by the weight decay term in neural networks. It is desirable to have smooth, broadband 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 DNNs is shown in Fig. 13. Based on this curve, we chose a weight decay term of 2×10^{-3} for all our experiments.

5.8 Binary classification

The simplest task possible with our optical setup is a binary classification, where the label at each pixel belongs to one of

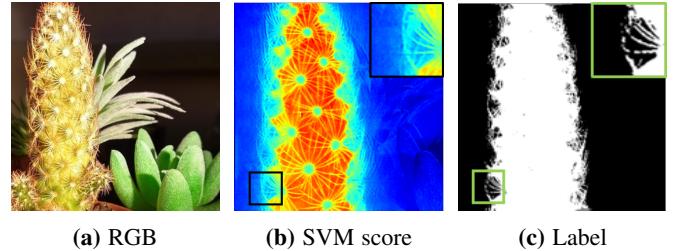


Fig. 14: **Per-pixel classification.** Due to per-pixel operation with high spatial resolution, our imager can clearly identify the micro-structures such as the cactus thorns by capturing only two images instead of the complete HSI.

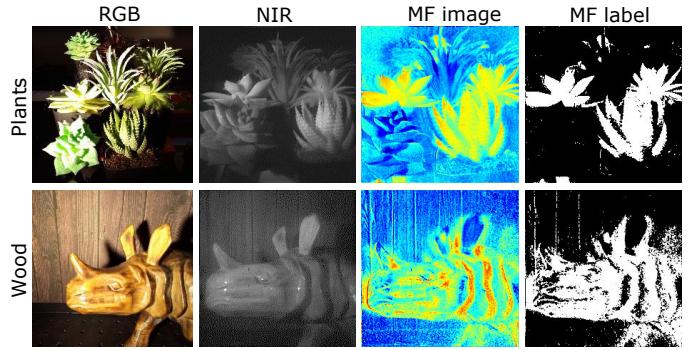


Fig. 15: **Matched filtering.** If spectrum of the specific objects is known, one can implement a matched filter and perform per-pixel material classification.

the two possible classes. In such a situation, one may either use a linear SVM where the spectral filter is the learned supporting hyperplane, 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 14 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 15 shows classification results for real vs. plastic plants and real vs. fake wood with matched filtering and Fig. 16 shows classification results with SVM classifier.

Figure 17 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) curve of classification performance for both cases. The SNR advantage is evident; the area under the curve for optical projection (0.7914) is higher than full measurement and then projection (0.7912).

5.9 Remote photoplethysmography

By programming our 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 19 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

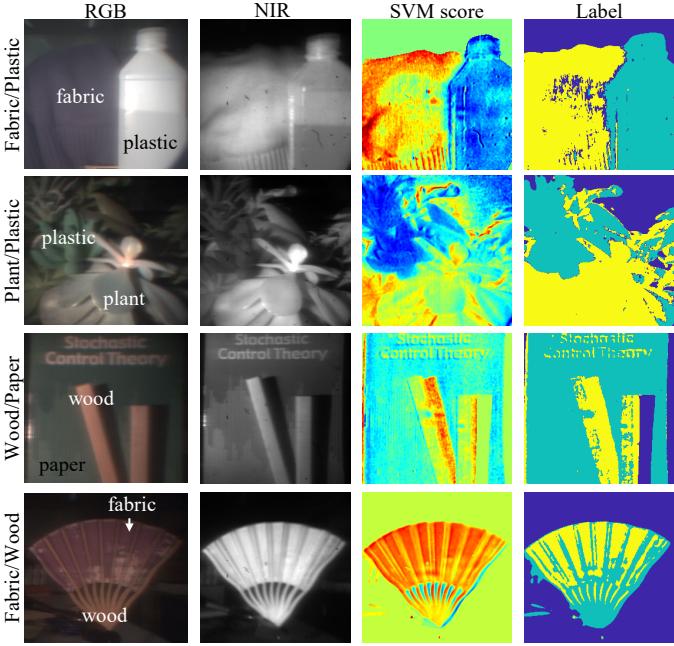


Fig. 16: SVM classification. We learned SVM filters on spectrometer training data and then implemented them on our optical setup. Our setup is very versatile and can be used for classifying arbitrary material classes.

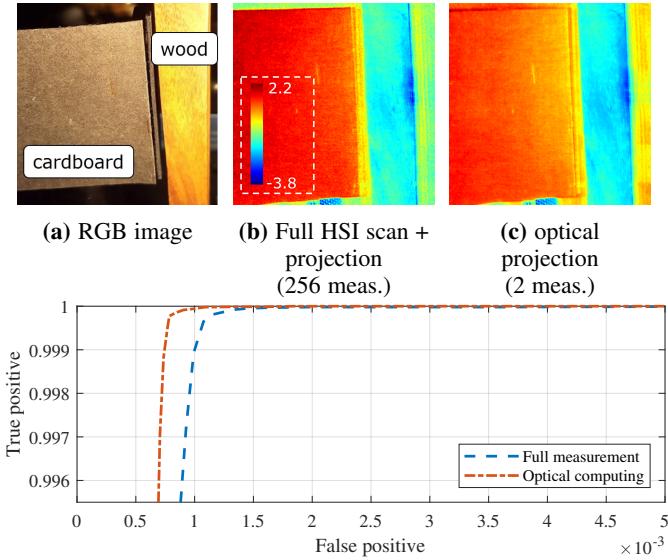


Fig. 17: Advantage of optical computing. We show an example of binary classification between cardboard and wood (a) using per-pixel SVM. Optical computing achieves higher accuracy with far fewer measurements.

this signal was 80.7 beats per minute, while an off-the-shelf pulse oximeter reported 79 – 84 beats per minute.

5.10 Multi-class classification with DNNs

We first tested our prototype on an outdoor scene shown in Fig. 20 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.

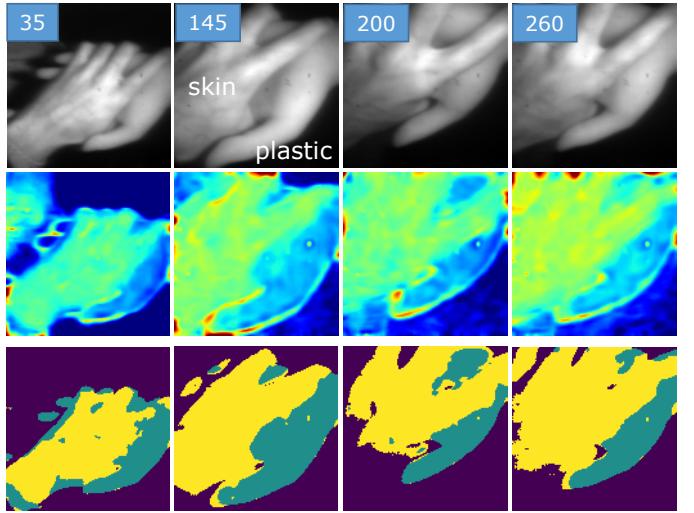


Fig. 18: Video rate classification. Since our setup can rapidly change spectral filters, we can implement video-rate classification. In this example, we show some frames of a classifier that separates human skin from plastic (silicone in this example), which is often useful for biometrics.

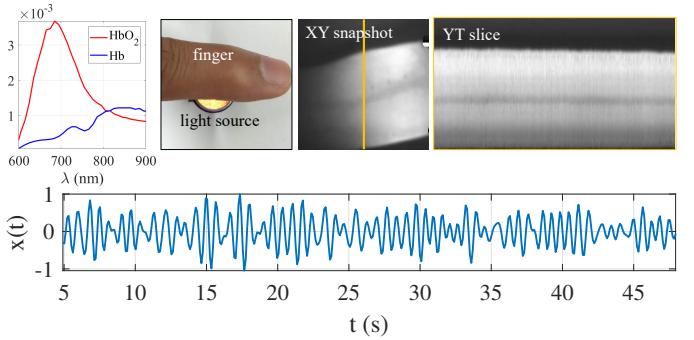


Fig. 19: Remote pulse oximetry. Hemoglobin in blood varies between being oxygenated and deoxygenated, thereby having different spectral profiles (left plot). We program our camera with a difference of the two profiles, to capture oximetric signal in a remote, non-contact fashion. The plot in the second row shows the processed oximetry signal over the finger.

The resultant label map is shown in Fig. 20. 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. 11) and classified several indoor objects. Figure 21 compares the spectrally filtered output using full scan, as well as optical projection. The two results look very similar; while optical computing requires only five images, full scanning required 256 images. Figure 22 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 all-pass spectral filter.

5.11 Discussion

Across all the experiments, we note that the proposed optical classification strategy is promising, particularly when operated in binary classification mode. 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



Fig. 20: **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.

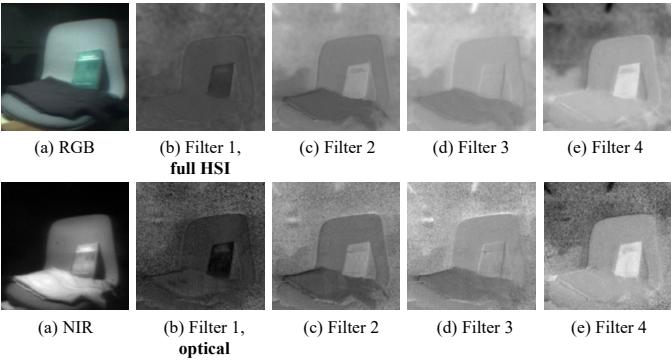


Fig. 21: **Comparison of optical and full-scan spectral filtering.** Our method directly measures the spectral projections, which requires 5 images instead of 256 images (full HSI scanning).

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. Second, higher order effects such as non-Lambertian surfaces, global illuminant component and scatter cause the measured spectrum to be different than 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.

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.

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

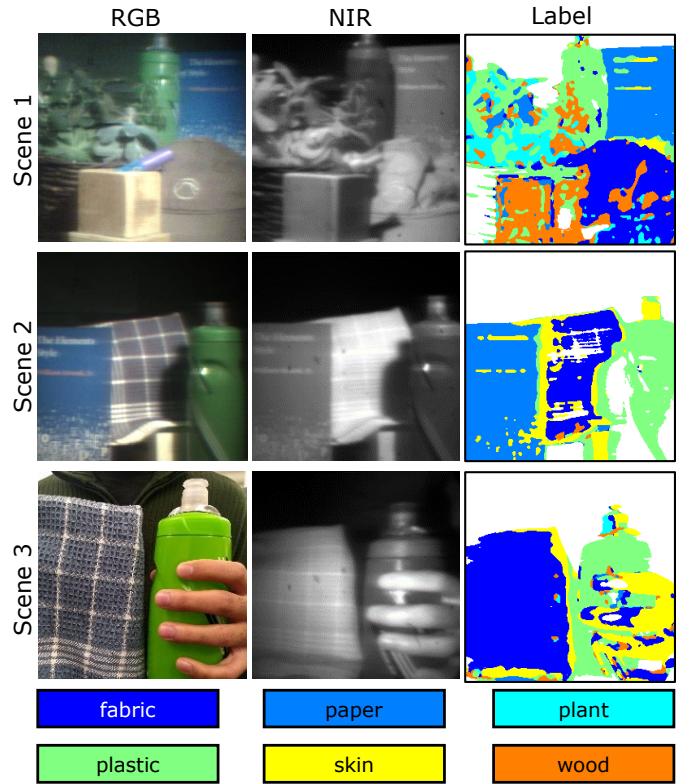


Fig. 22: **Multiclass classification with DNNs.** We show diverse set of scenes that were classified with our optical classification strategy.

everyday materials which we believe will enable future research in spectrum-based material classification for computer vision.

7 ACKNOWLEDGMENTS

The authors thank Ms. Anqi Yang for helping with data collection. The authors acknowledge support via the National Geospatial-Intelligence Agency's Academic Research Program (Award No. HM0476-17-1-2000), the NSF SaTC award 1801382, and the NSF Expeditions award 1730147.

REFERENCES

- [1] Programmable-spectrometry code. <https://github.com/image-science-lab/Programmable-Spectrometry>.
- [2] R. G. Baraniuk. Compressive sensing. *IEEE Signal Processing Magazine*, 24(4):118–121, 2007.
- [3] A. Chakrabarti. Learning sensor multiplexing design through back-propagation. In *Adv. in Neural Info. Processing Systems*, 2016.
- [4] H. G. Chen, S. Jayasuriya, J. Yang, J. Stephen, S. Sivaramakrishnan, A. Veeraraghavan, and A. Molnar. ASP Vision: Optically computing the first layer of convolutional neural networks using angle sensitive pixels. In *IEEE Intl. Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [5] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi. Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *Trans. Geoscience and Remote Sensing*, 54(10):6232–6251, 2016.
- [6] E. Cloutis. Review article: Hyperspectral geological remote sensing: Evaluation of analytical techniques. *International J. Remote Sensing*, 17(12):2215–2242, 1996.
- [7] N. Colthup. *Introduction to Infrared and Raman Spectroscopy*. Elsevier, 2012.
- [8] M. Dalponte, L. Bruzzone, L. Vescovo, and D. Ganelle. The role of spectral resolution and classifier complexity in the analysis of hyperspectral images of forest areas. *Remote Sensing of Environment*, 113(11):2345–2355, 2009.

- [9] N. Dobigeon, J.-Y. Tourneret, C. Richard, J. C. M. Bermudez, S. McLaughlin, and A. O. Hero. Nonlinear unmixing of hyperspectral images: Models and algorithms. *IEEE Signal Processing Magazine*, 31(1):82–94, 2014.
- [10] Z. Erickson, N. Luskey, S. Chernova, and C. C. Kemp. Classification of household materials via spectroscopy. *IEEE Robotics and Automation Letters*, 4(2):700–707, 2019.
- [11] M. Fauvel, J. Chanussot, J. A. Benediktsson, and J. R. Sveinsson. Spectral and spatial classification of hyperspectral data using SVMs and morphological profiles. In *IEEE Intl. Geoscience and Remote Sensing*, 2007.
- [12] M. Goel, E. Whitmire, A. Mariakakis, T. S. Saponas, N. Joshi, D. Morris, B. Guenter, M. Gavriliu, G. Borriello, and S. N. Patel. Hypercam: Hyperspectral imaging for ubiquitous computing applications. In *ACM Intl. Joint Conf. Pervasive and Ubiquitous Computing*, 2015.
- [13] R. Gross, I. Matthews, and S. Baker. Fisher light-fields for face recognition across pose and illumination. In *Joint Pattern Recognition Symposium*, 2002.
- [14] A. B. Hamida, A. Benoit, P. Lambert, and C. B. Amar. 3-d deep learning approach for remote sensing image classification. *Trans. Geoscience and Remote Sensing*, 56(8):4420–4434, 2018.
- [15] J. C. Harsanyi and C.-I. Chang. Hyperspectral image classification and dimensionality reduction: An orthogonal subspace projection approach. *IEEE Trans. Geoscience and Remote Sensing*, 32(4):779–785, 1994.
- [16] M. He, B. Li, and H. Chen. Multi-scale 3d deep convolutional neural network for hyperspectral image classification. In *IEEE Intl. Conf. Image Processing*, 2017.
- [17] J. B. Hearnshaw. *The Analysis of Starlight: Two Centuries of Astronomical Spectroscopy*. Cambridge University Press, 2 edition, 2014.
- [18] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li. Deep convolutional neural networks for hyperspectral image classification. *J. Sensors*, 2015.
- [19] M. H. Kim, T. A. Harvey, D. S. Kittle, H. Rushmeier, J. Dorsey, R. O. Prum, and D. J. Brady. 3d imaging spectroscopy for measuring hyperspectral patterns on solid objects. *ACM Trans. Graphics*, 31(4):38, 2012.
- [20] H. Lee and H. Kwon. Contextual deep cnn based hyperspectral classification. In *Intl. Geoscience and Remote Sensing*, 2016.
- [21] C. Li, T. Sun, K. F. Kelly, and Y. Zhang. A compressive sensing and unmixing scheme for hyperspectral data processing. *IEEE Trans. Image Processing*, 21(3):1200–1210, 2012.
- [22] Y. Li, H. Zhang, and Q. Shen. Spectral-spatial classification of hyperspectral imagery with 3d convolutional neural network. *Remote Sensing*, 9(1):67, 2017.
- [23] J. W. Lichtman and J.-A. Conchello. Fluorescence microscopy. *Nature Methods*, 2(12):910, 2005.
- [24] B. Liu, X. Yu, P. Zhang, X. Tan, A. Yu, and Z. Xue. A semi-supervised convolutional neural network for hyperspectral image classification. *Remote Sensing*, 8(9):839–848, 2017.
- [25] C. Liu and J. Gu. Discriminative illumination: Per-pixel classification of raw materials based on optimal projections of spectral BRDF. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 36(1):86–98, 2014.
- [26] S. P. Love and D. L. Graff. Full-frame programmable spectral filters based on micromirror arrays. *J. Micro/Nanolithography, MEMS, and MOEMS*, 13(1):011108, 2014.
- [27] Y. Luo, J. Zou, C. Yao, X. Zhao, T. Li, and G. Bai. Hsi-cnn: A novel convolution neural network for hyperspectral image. In *Intl. Conf. Audio, Language and Image Processing*, 2018.
- [28] F. Melgani and L. Bruzzone. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans. Geoscience and Remote Sensing*, 42(8):1778–1790, 2004.
- [29] A. Mohan, R. Raskar, and J. Tumblin. Agile spectrum imaging: Programmable wavelength modulation for cameras and projectors. In *Computer Graphics Forum*, 2008.
- [30] S. K. Nayar and R. M. Bolle. Computing reflectance ratios from an image. *Pattern Recognition*, 26(10):1529–1542, 1993.
- [31] S. K. Nayar and R. M. Bolle. Reflectance ratio: A photometric invariant for object recognition. In *IEEE Intl. Conf. Computer Vision (ICCV)*, 1993.
- [32] S. K. Nayar and R. M. Bolle. Reflectance based object recognition. *Intl. J. Comp. Vision*, 17(3):219–240, 1996.
- [33] M. O’Toole and K. N. Kutulakos. Optical computing for fast light transport analysis. *ACM Trans. Graphics*, 29(6):164, 2010.
- [34] Z. Pan, G. Healey, M. Prasad, and B. Tromberg. Face recognition in hyperspectral images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 25(12):1552–1560, 2003.
- [35] J.-I. Park, M.-H. Lee, M. D. Grossberg, and S. K. Nayar. Multispectral imaging using multiplexed illumination. In *IEEE Intl. Conf. Computer Vision (ICCV)*, 2007.
- [36] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. In *Ad. in Neural Info. Processing Systems Workshop*, 2017.
- [37] 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. Scikit-learn: Machine learning in Python. *J. Machine Learning Research*, 12:2825–2830, 2011.
- [38] A. Ramirez, G. R. Arce, and B. M. Sadler. Coded-aperture compressive spectral image classification. In *Computational Optical Sensing and Imaging*, 2012.
- [39] A. Ramirez, G. R. Arce, and B. M. Sadler. Hyperspectral pixel classification from coded-aperture compressive imaging. In *Independent Component Analyses, Compressive Sampling, Wavelets, Neural Net, Biosystems, and Nanoengineering X*, 2012.
- [40] A. Ramirez, H. Arguello, G. R. Arce, and B. M. Sadler. Spectral image classification from optimal coded-aperture compressive measurements. *IEEE Trans. Geoscience and Remote Sensing*, 52(6):3299–3309, 2013.
- [41] J. P. Rice, S. W. Brown, J. E. Neira, and R. R. Bousquet. A hyperspectral image projector for hyperspectral imagers. In *Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XIII*, 2007.
- [42] N. Salamati, C. Fredembach, and S. Süssstrunk. Material classification using color and nir images. In *Color and Imaging Conference*, 2009.
- [43] V. Saragadam and A. Sankaranarayanan. On space-spectrum uncertainty analysis for spectrally programmable cameras. *Optics Express*, 28(6):7771–7785, 2020.
- [44] V. Saragadam and A. C. Sankaranarayanan. KRISM—Krylov subspace-based optical computing of hyperspectral images. *ACM Trans. Graphics*, 38(5):148:1–14, 2019.
- [45] V. Sharma, A. Diba, T. Tuytelaars, and L. Van Gool. Hyperspectral cnn for image classification and band selection, with application to face recognition. *Technical report KUL/ESAT/PSI/1604, KU Leuven, ESAT, Leuven, Belgium*, 2016.
- [46] Y. Tarabalka, J. Chanussot, and J. A. Benediktsson. Segmentation and classification of hyperspectral images using watershed transformation. *Pattern Recognition*, 43(7):2367–2379, 2010.



Vishwanath Saragadam received his B.Tech degree in Electrical Engineering from Indian Institute of Technology Madras in 2014, where he won the Siemens’s award for highest GPA in Electrical Engineering. He obtained his MS in Electrical and Computer Engineering from Carnegie Mellon University in 2016, where he is currently pursuing PhD. His research interests include sparse approximation techniques, hyperspectral imaging, computational photography, compressive sensing and computer vision. He received the Outstanding TA award from the ECE Department at CMU in 2018.



Aswin C. Sankaranarayanan (SM’17) is an Associate Professor in the ECE Department at Carnegie Mellon University (CMU). He earned his Ph.D. from University of Maryland, College Park and was a post-doctoral researcher at the DSP group at Rice University before joining CMU. Aswin’s research spans topics in imaging, vision, and image processing. He is the recipient of the CVPR 2019 best paper award, the CIT Dean’s Early Career Fellowship in 2018, the NSF CAREER award in 2017, the Eta Kappa Nu (Sigma chapter) Excellence in Teaching award in 2017, the 2016 Hershel M. Rich invention award, and the distinguished dissertation fellowship from the ECE Department at University of Maryland in 2009.