

Article

Multitrack Compressed Sensing for Faster Hyperspectral Imaging

Sharvaj Kubal ^{1,2,3} , Elizabeth Lee ² , Chor Yong Tay ^{3,4}  and Derrick Yong ^{1,2,*} 

- ¹ Singapore Institute of Manufacturing Technology, Agency for Science, Technology and Research, 2 Fusionopolis Way, Singapore 138634, Singapore; sharvaj.kubal@smart.mit.edu
- ² Critical Analytics for Manufacturing Personalized-Medicine, Singapore-MIT Alliance for Research and Technology Centre, 1 Create Way, Singapore 138602, Singapore; elizabeth.lee@smart.mit.edu
- ³ School of Materials Science and Engineering, Nanyang Technological University, 50 Nanyang Ave., Singapore 639798, Singapore; cytay@ntu.edu.sg
- ⁴ School of Biological Sciences, Nanyang Technological University, 50 Nanyang Ave., Singapore 639798, Singapore
- * Correspondence: derrick-yong@SIMTech.a-star.edu.sg

Abstract: Hyperspectral imaging (HSI) provides additional information compared to regular color imaging, making it valuable in areas such as biomedicine, materials inspection and food safety. However, HSI is challenging because of the large amount of data and long measurement times involved. Compressed sensing (CS) approaches to HSI address this, albeit subject to tradeoffs between image reconstruction accuracy, time and generalizability to different types of scenes. Here, we develop improved CS approaches for HSI, based on parallelized multitrack acquisition of multiple spectra per shot. The multitrack architecture can be paired up with either of the two compatible CS algorithms developed here: (1) a sparse recovery algorithm based on block compressed sensing and (2) an adaptive CS algorithm based on sampling in the wavelet domain. As a result, the measurement speed can be drastically increased while maintaining reconstruction speed and accuracy. The methods were validated computationally both in noiseless as well as noisy simulated measurements. Multitrack adaptive CS has a ~10 times shorter measurement plus reconstruction time as compared to full sampling HSI without compromising reconstruction accuracy across the sample images tested. Multitrack non-adaptive CS (sparse recovery) is most robust against Poisson noise at the expense of longer reconstruction times.



Citation: Kubal, S.; Lee, E.; Tay, C.Y.; Yong, D. Multitrack Compressed Sensing for Faster Hyperspectral Imaging. *Sensors* **2021**, *21*, 5034. <https://doi.org/10.3390/s21155034>

Academic Editor: Benoit Voziel

Received: 24 June 2021

Accepted: 20 July 2021

Published: 24 July 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: hyperspectral imaging; compressed sensing; wavelets; adaptive imaging

1. Introduction

Hyperspectral images are comprised of light intensity information from an object or scene, resolved into two spatial dimensions and a spectral (wavelength) dimension, forming a datacube. Such images can be interpreted as a collection of (i) electromagnetic spectra associated with each spatial pixel or (ii) grayscale images corresponding to each of the spectral channels.

Hyperspectral imaging (HSI)—the capture of hyperspectral images—proves beneficial in areas as diverse as biomedicine [1], materials defect detection [2], agriculture monitoring and food safety [3], and even beyond more traditional applications, such as remote sensing of terrain [4]. Variants of HSI, such as Raman chemical imaging [5], where each spatial pixel records a Raman spectrum, are useful as label-free compositional analysis methods for spatially heterogeneous samples. Such tools are valuable in biopharmaceutical and cell manufacturing.

The biopharmaceutical industry has been trending towards integrated continuous manufacturing [6], where real-time measurements and process control [7] are critical in ensuring and optimizing the quality of the products. This is particularly important due

to the stringent regulatory landscape. Here, optical techniques such as HSI can provide information-rich yet non-contact methods for monitoring and inspection.

Hyperspectral imaging presents a challenge due to the large amount of data that has to be acquired (and subsequently stored) and the long duration of time required to acquire this data. Established methods that fully sample the hyperspectral scene include point-scan, line-scan (pushbroom), wavelength-scan and snapshot imaging. (For clarity in description, we distinguish between the terms hyperspectral *scene* and hyperspectral *image*. The former will refer to the object or scene that is *to be imaged* hyperspectrally; the latter will refer to the datacube captured by an imager.) Scanning methods can be time-consuming, especially in low-light conditions where long integration times are needed. These methods also often involve moving parts. Furthermore, if the scene to be imaged is dynamic, scanning can produce artifacts that are difficult to correct during postprocessing. On the other hand, snapshot imaging methods, which record the whole image in one shot, involve very large detector arrays and complex optical arrangements such as lenslet arrays and filter arrays [8]. Overall, there is a necessity for subsampling (or undersampling) approaches to HSI such that measurements can be made faster and more efficient without losing critical information in the process.

A standard approach to the subsampling of signals is compressed sensing (CS), also known as compressive sensing or compressive sampling [9,10]. The theory of compressed sensing prescribes a subsampling procedure, together with a reconstruction algorithm, which recovers the signal (the datacube in our case) by making use of some prior assumptions about the structure of the signal. Recent HSI approaches have been leveraging on this theory in a variety of ways [11]. Coded aperture snapshot spectral imaging (CASSI) [12–14] provides a prominent class of imaging architectures for compressed sensing HSI (CS-HSI), that can achieve a snapshot measurement—capturing all the necessary data in one shot instead of using a sequence of measurements. Another important HSI architecture is the single pixel camera (SPC)-style hyperspectral imager [15], which is a direct extension of the SPC by replacing the single pixel detector with a spectrometer. There have also been recent efforts in making CS-based hyperspectral imagers more compact. The use of spatial light modulators or coded apertures is forgone in [16,17], and the number of optical elements required is greatly reduced.

Classes of CS algorithms commonly employed are sparse recovery [18,19] (which can be augmented by dictionary learning [20] or learnt nonlinear representations [21]), neural network-based reconstruction [22–25] and adaptive basis scan (or adaptive direct sampling) [26–29]. Sparse recovery algorithms tend to have good performance guarantees but are generally computationally intensive, making them slow for large signals such as hyperspectral scenes. Neural network-based reconstruction, on the other hand, is much faster. Neural networks use training signals to learn an inverse transformation from the subsampled measurements back to the original signal, hence avoiding slow optimization steps post-training. However, a lack of sufficient training data or mismatches between the distributions of the training set and the set of target signals can limit the generalizability and scope of this method. Lastly, an adaptive basis scan is based on progressive prediction and sampling of the most significant coefficients of a signal in some basis. These prediction steps are computationally lighter than the iterative steps involved in sparse recovery, and as a result, adaptive basis scan also achieves fast reconstruction speeds. Moreover, it is not limited by the need for large training datasets, unlike neural-network-based methods.

In the context of contemporary manufacturing, where there is an increasing need for high-volume production with extensive personalization [30], we require responsive yet autonomous operations together with end-to-end integration of the various processes. The monitoring and inspection systems involved in such a setting would require real-time processing and feedback while maintaining reliability, robustness and consistency [31].

CASSI is not very suitable for these purposes since it cannot be used with an adaptive basis scan (to our knowledge); it fails the ‘fast reconstruction’ criterion when using sparse recovery and the ‘generalizability’ criterion when using neural-networks for reconstruction.

The alternative here, the SPC-style hyperspectral imager, is also limited in its capability as it only measures one spectrum in a shot. Hence, a large number of shots are needed to obtain sufficient information for a good reconstruction of the hyperspectral datacube. As a consequence, the measurement times are high despite the fact that reconstruction times can be reduced by computationally light algorithms. Overall, there is a need for HSI methods that can achieve fast measurement as well as reconstruction while maintaining imaging accuracy and generalizability across various types of scenes.

Here, we formulate improved compressed sensing approaches for HSI, which overcome the aforementioned limitations of measurement and reconstruction speeds. This is achieved by multitrack compressed sensing, based on a multitrack acquisition architecture enabling multiple spectra to be measured in each shot. Besides being compatible with standard sparse recovery techniques, multitrack compressed sensing also supports an adaptive basis scan so that multiple basis coefficients can be sampled in parallel. As a result, the measurement speed can be drastically improved while maintaining reconstruction speed as well as accuracy.

The paper is organized as follows. Section 2 reviews the prior work on compressed sensing for HSI, describes the proposed multitrack acquisition architecture and develops the two compatible CS algorithms for it: a non-adaptive sparse-recovery-based algorithm and an adaptive basis scan algorithm. Section 3 describes the numerical experiments conducted and discusses the reconstruction results for noiseless as well as noisy measurements. Finally, Section 4 concludes.

2. Materials and Methods

2.1. Background

2.1.1. CS Formulation and Notation

Here, we describe the general framework for compressed sensing [9,10]. Let $x \in \mathbb{R}^n$ (represented by a column vector) be a discrete signal that is to be measured. The measurement (alternatively sampling or acquisition) of x is described by the equation:

$$y = \Phi x \quad (1)$$

where $y \in \mathbb{R}^m$ is a column vector whose components are all the measurement values, and $\Phi \in \mathbb{R}^{m \times n}$ is the measurement matrix. An individual measurement y_j can be written as a dot product (or projection) $y_j = \varphi_j^T x$ between the signal and the j th row φ_j^T of Φ . Subsampling occurs when the number of measurements m is less than the length of the signal n , i.e., when the sampling ratio $r_{\text{sampling}} = m/n$ is less than 1. After the subsampling process, the original signal x is reconstructed from the measurements y by methods such as sparse recovery.

The particular type of signals addressed in this paper is hyperspectral scenes (and images), which can be naturally organized into datacubes. Mathematically, this is a 3D array of the shape $n_h \times n_w \times n_c$, where n_h is the spatial height of the scene or image in pixels, n_w is the spatial width in pixels, and n_c is the number of wavelength channels. $n_p \equiv n_h n_w$ denotes the total number of spatial pixels. It is common and convenient for use in later equations, however, to reshape the elements of the datacube into a matrix: $X \in \mathbb{R}^{(n_h n_w) \times n_c}$, whose row indices correspond to spatial pixel locations and column indices correspond to wavelength channels. This is performed by taking spatial slices of the datacube at each wavelength channel and unrolling these slices into tall column vectors, which end up forming the columns of X .

Note that the symbol X here plays the same role as the symbol x in Equation (1)—the only difference is that the elements of X are arranged into a matrix instead of a column vector. This notation will be used in the following sections of the paper.

2.1.2. Acquisition Architectures

Several CS acquisition architectures that have been employed [11] make use of devices that can modulate and code light spatially. Spatial light modulators include masks [32],

which block out light in some regions and not in others, and digital micromirror devices (DMD) [33], which are arrays of micrometer-scale mirrors that can dynamically form patterns that selectively redirect light from different regions.

Coded aperture snapshot spectral imaging (CASSI) [12–14] provides a prominent class of imaging architectures for CS-HSI, which can achieve snapshot HSI. The snapshot nature enables capturing hyperspectral video at high frame rates [34]. However, such powerful architectures can come with increased hardware costs—to image an n_h (height)-by- n_w (width)-by- n_c (wavelength channels) voxel scene using single dispersive CASSI, for example, the detector array needs to have dimensions of at least $(n_h) \times (n_w + n_c - 1)$ pixels. The number $n_w + n_c - 1$ can generally be very large, and this additionally presents a trade-off between the amount of spatial and spectral information that can be captured at once. Dual dispersive CASSI can function with a detector array having fewer pixels, but there is still the issue that significantly low amounts of spectral information can be captured in one shot. Multiframe CASSI [35] acquires more information through multiple shots, each using a distinct mask or DMD pattern, albeit the snapshot nature of the method is inevitably lost. Hence, there exists a different trade-off: one between spectral information and acquisition time. Furthermore, CASSI systems use prisms as the dispersive element and since the dispersion of a prism is nonlinear, the spectral resolution of CASSI varies with the wavelength [36].

High-accuracy and high-resolution spectral data warrant the use of a spectrometer. The single pixel camera (SPC)-style hyperspectral imager [15] is an acquisition architecture that leverages this—it consists of an SPC with the single pixel detector replaced by a spectrometer. Being an extension of the SPC, this architecture allows us to borrow many of the CS methods developed originally for single pixel imaging, including an adaptive basis scan. The capability of the architecture to sample dot products of patterns on the DMD with the spatial slices of the datacube allows for direct sampling of the wavelet coefficients of the scene. Note, however, that unlike snapshot CASSI, the SPC-style imager requires sequential measurements of spectra through a large number of shots. Therefore, even though faster adaptive reconstruction techniques are enabled, the slow measurement process nullifies this advantage. Addressing this limitation is the central theme of this work.

2.1.3. Adaptive Basis Scan Algorithms

Adaptive basis scan was briefly described in Section 1. It is based on progressive prediction and sampling of the most significant coefficients of signals in some basis. The most popular bases for this method are wavelets [37]. Wavelet transforms decompose signals into coefficients that capture the signals' features at multiple scales and locations. Such multiresolution analyses provide the transform coefficients with a hierarchical structure so that some coefficients have predictive power over others. In particular, low-resolution information that is contained in the wavelet coefficients at coarser scales can be used to predict which of the wavelet coefficients at finer scales are significant.

Deutsch et al. [26] predicted the significance of the fine-scale coefficients by estimating local directional Lipschitz exponents from the already-sampled coarse-scale coefficients. Averbuch et al. [27], on the other hand, relied on statistical modeling of the wavelet coefficients and their parent-child relationships. In [38], Hahn et al. adapted Deutsch's adaptive algorithm to hyperspectral imaging. Finally, Rousset et al. [29] developed an adaptive basis scan algorithm whose wavelet coefficient prediction step relies on bicubic interpolation in image space. This was further extended to time-resolved HSI in [39]. In [38,39], however, the measurement step is not multitrack—the sensor elements measure only a single spectrum per shot.

The multitrack adaptive CS algorithm we present in Section 2.2.3 is an extension of the Adaptive Basis Scan with Wavelet Prediction (ABS-WP) algorithm from [29,39]. We make this choice because ABS-WP produces high accuracy reconstruction, even in the presence of noise. Furthermore, ABS-WP has already been demonstrated to work with

HSI via minor changes to the original algorithm. However, multitrack CS is potentially compatible with the other adaptive basis scan algorithms as well.

2.2. Proposed Multitrack Compressed Sensing

The single-track acquisition of the SPC-style hyperspectral imager measures only one spectrum per shot, and hence, a large number of shots are needed to obtain sufficient information for a good reconstruction of the hyperspectral scene. This is especially a problem when integration times are large, as is the case for low-light imaging. To develop more powerful acquisition architectures, one can draw inspiration from pushbroom hyperspectral imagers in which multiple spectra are measured at once.

2.2.1. Multitrack Acquisition Architecture

Consider the hyperspectral scene $X \in \mathbb{R}^{n_p \times n_c}$, whose components have units of photons per second. Different spatial regions—rectangular tracks (Figure 1a)—of the hyperspectral scene are to be treated separately yet in parallel. Let n_τ be the number of tracks. Then, each track of the scene will be n_h -by- (n_w/n_τ) -by- n_c in terms of voxels. Spatially, each track has $n_h \times (n_w/n_\tau) = n_p/n_\tau$ pixels. In our matrix notation of X , this division into tracks can be described by breaking down X into submatrices:

$$X = \begin{bmatrix} X^{(1)} \\ \vdots \\ X^{(n_\tau)} \end{bmatrix} \quad (2)$$

where $X^{(1)}, \dots, X^{(n_\tau)} \in \mathbb{R}^{(n_p/n_\tau) \times n_c}$ represent the tracks. Next, the different tracks of X need to be modulated separately but in parallel. This is achieved by dividing the DMD into tracks of size n_h -by- (n_w/n_τ) as well to correspond spatially to the tracks of the hyperspectral scene (Figure 1b). The pixels on the i th track of the DMD form the i th subpattern $\zeta^{(i)} \in \mathbb{R}^{(n_p/n_\tau)}$. This modulates the i th track of X , i.e., $X^{(i)}$. All tracks of X simultaneously undergo such modulation, after which the entire modulated scene is focused into a line by a cylindrical lens (Figure 1c). Different sections of the line are comprised of information from different tracks. This can be modeled by a trackwise list of dot products between subpatterns and the corresponding tracks of X :

$$\begin{bmatrix} (\zeta^{(1)})^T X^{(1)} \\ \vdots \\ (\zeta^{(n_\tau)})^T X^{(n_\tau)} \end{bmatrix} \in \mathbb{R}^{n_\tau \times n_c} \quad (3)$$

where each $(\zeta^{(i)})^T X^{(i)} \in \mathbb{R}^{1 \times n_c}$. Finally, the transformed scene enters a spectrometer where the focused line is dispersed. The 2D detector array captures one spectrum from each track, i.e., n_τ spectra in total. With integration time Δt , the recorded spectra in a given shot (in the units of photon counts) are described by the rows of:

$$\begin{bmatrix} (\zeta^{(1)})^T X^{(1)} \Delta t \\ \vdots \\ (\zeta^{(n_\tau)})^T X^{(n_\tau)} \Delta t \end{bmatrix} \in \mathbb{R}^{n_\tau \times n_c} \quad (4)$$

assuming no measurement noise. Overall, one shot of the multitrack acquisition process comprises of the simultaneous (or parallel) action of subpatterns $\zeta^{(1)}, \dots, \zeta^{(n_\tau)}$ to linearly map X onto the recorded spectra $(\zeta^{(i)})^T X^{(i)} \Delta t$.

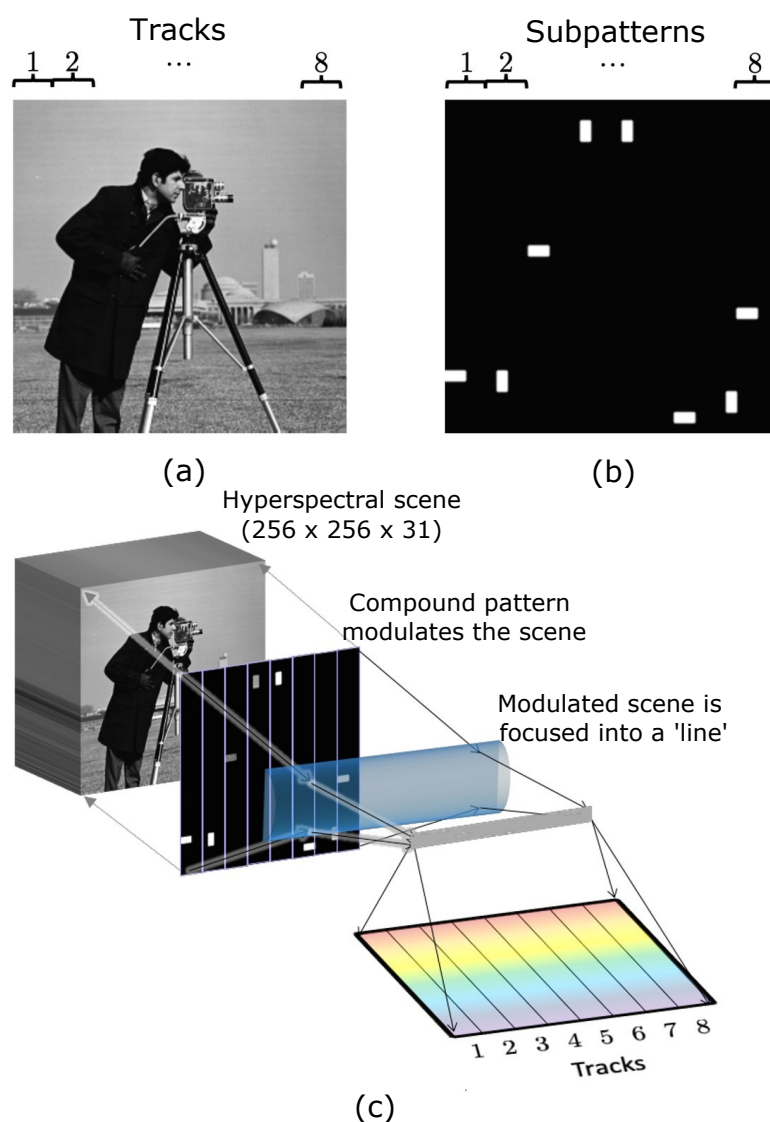


Figure 1. (a) Dividing a hyperspectral scene into tracks (8 tracks as an example) seen here as the dashed vertical strips. (b) A full compound pattern as formed by subpatterns (8 subpatterns in this example), again seen as the dashed vertical strips. (c) A schematic of the multitrack acquisition process. In every shot, the hyperspectral scene is modulated by a compound pattern, focused into a line and dispersed in a spectrometer. The 2D detector array records one spectrum from each track.

2.2.2. Multitrack Non-Adaptive Compressed Sensing

This section describes the first of our proposed methods. It is based upon block compressed sensing and uses non-adaptive sparse recovery for reconstruction. Block compressed sensing [40,41] involves dividing a scene into blocks and applying the measurement matrix to those blocks separately. The reconstruction can then be carried out either separately for each block or globally by processing all the blocks together. The aim of block CS is to reduce the reconstruction complexity and the memory burden of the imaging process.

The track-wise acquisition naturally allows for the use of block compressed sensing methods, where the tracks can be treated as blocks that have their own measurement matrices. The reconstruction algorithm then takes into account this measurement matrix structure, as will be described later.

Note that a non-adaptive compressed sensing HSI architecture with a similar multitrack acquisition principle was first developed, to our knowledge, in [42]. Per shot, a spectrum is captured from each pixel-wide column of the scene; this is the same as having

one-pixel-wide tracks in our formulation from Section 2.2.1. The use of block compressed sensing provides a generalization to this, allowing the tracks to have a width greater than one pixel. This is important when the number of spectra that can be measured per shot is limited by the detector pixel count. Several spectrometers have 2D detector arrays of 256-by-1024 pixels, for example, where a 1-by-1024 strip is meant for a single spectrum. In that case, the maximum number of spectra that can be measured in a shot is 256. Moreover, due to optical aberrations near the edges of the detector array, the maximum number of spectra might be even lower in practice. This is a limiting factor when the desired datacube is more than 256 pixels wide. Our use of block CS, however, enables the use of wider tracks. As a result, fewer tracks can be used cover the whole datacube, and fewer spectra need to be measured per shot.

First, we describe the measurement process. As separate tracks are measured separately, we can rewrite Equation (1) for each track:

$$Y^{(i)} = \Phi^{(i)} X^{(i)} \quad \text{for each track } i \quad (5)$$

where $\Phi^{(i)} \in \mathbb{R}^{m^{(i)} \times (n_p/n_\tau)}$ is track i 's measurement matrix, and $m^{(i)}$ is the number of samples taken from track i , which is related to the sampling ratio by $m^{(i)} = r_{\text{sampling}} \times n_p/n_\tau$. $Y^{(i)} \in \mathbb{R}^{m^{(i)} \times n_c}$ are the measured spectra from the track (up to some scaling factors). The measurement matrices $\Phi^{(i)}$ we use are composed of entries drawn from a transformed Bernoulli distribution: $\pm 1/\sqrt{m^{(i)}}$ with equal probabilities. Here $\sqrt{m^{(i)}}$ works as a normalization.

Now, we describe how the measurement matrices can be implemented using a DMD. Consider track i . The k th projection from this track can be written as the dot product

$$Y_{k,\cdot}^{(i)} = \left(\varphi_k^{(i)} \right)^T X^{(i)} \quad (6)$$

where $Y_{k,\cdot}^{(i)}$ denotes the k th row of $Y^{(i)}$ (equivalent MATLAB-like notation would be $Y^{(i)}(k, :)$), and $\left(\varphi_k^{(i)} \right)^T$ is the k th row of $\Phi^{(i)}$. To implement this projection, we employ the following scaling and shifting operations on $\varphi_k^{(i)}$ and obtain physically implementable b -bit (grayscale) subpatterns:

$$\zeta_k^{(i)} = \frac{2^{b-1} - 1}{2^b - 1} \left(\frac{\varphi_k^{(i)}}{\|\varphi_k^{(i)}\|_\infty} + \mathbf{1}_{n_p}^{(i)} \right) \quad \text{for all tracks } i \quad (7)$$

Here, $\|\cdot\|_\infty$ denotes the ℓ_∞ -norm, and $\mathbf{1}_{n_p}^{(i)}$ is a subpattern $[1, \dots, 1]^T \in \mathbb{R}^{(n_p/n_\tau)}$ of the all-ones pattern $\mathbf{1}_{n_p} \in \mathbb{R}^{n_p}$:

$$\mathbf{1}_{n_p} = \begin{bmatrix} \mathbf{1}_{n_p}^{(1)} \\ \vdots \\ \mathbf{1}_{n_p}^{(n_\tau)} \end{bmatrix}$$

An additional offset subpattern is also required to compensate for the scaling and shifting mentioned above:

$$\zeta_{\text{offset}}^{(i)} = \frac{2^{b-1} - 1}{2^b - 1} \mathbf{1}_{n_p}^{(i)} \quad \text{for all tracks } i \quad (8)$$

These operations are needed because $\varphi_k^{(i)}$ cannot be physically implemented in a direct manner. As the DMD pixels values are physically restricted to the interval $[0, 1]$, the $\pm 1/\sqrt{m^{(i)}}$ values of $\varphi_k^{(i)}$ cannot be realized on the DMD. Furthermore, b -bit DMD pixel

values are subject to the discretization $\left\{ \frac{0}{2^b-1}, \frac{1}{2^b-1}, \dots, \frac{2^b-1}{2^b-1} \right\}$, which needs to be accounted for while constructing the subpatterns.

Next, these subpatterns are formed on the DMD to obtain raw measurements parallel to each track. For the k th shot, we get:

$$G(k) = \begin{bmatrix} \left(\zeta_k^{(1)} \right)^T X^{(1)} \Delta t \\ \vdots \\ \left(\zeta_k^{(n_\tau)} \right)^T X^{(n_\tau)} \Delta t \end{bmatrix} \quad (9)$$

with integration time Δt . From the offset subpatterns, we obtain

$$G_{\text{offset}} = \begin{bmatrix} \left(\zeta_{\text{offset}}^{(1)} \right)^T X^{(1)} \Delta t \\ \vdots \\ \left(\zeta_{\text{offset}}^{(n_\tau)} \right)^T X^{(n_\tau)} \Delta t \end{bmatrix} \quad (10)$$

The raw measurement can be processed as follows, based on Equations (6)–(10) to obtain the elements of Y :

$$Y_{k,\ell}^{(i)} = \frac{2^b - 1}{2^{b-1} - 1} \frac{\|\varphi_k^{(i)}\|_\infty}{\Delta t} ([G(k)]_{i,\ell} - [G_{\text{offset}}]_{i,\ell}), \quad \text{for all } i, \ell \quad (11)$$

where $[G(k)]_{i,\ell}$ and $[G_{\text{offset}}]_{i,\ell}$ denote (i, ℓ) th entries of $G(k)$ and G_{offset} , respectively. Note that G_{offset} does not depend on the projection index k , and thus, it only needs to be measured once. The measured value can subsequently be reused for other values of k . However, in the case of noisy measurements, it would be preferable to measure G_{offset} a few more times, with an average taken in the end.

Now, we describe the reconstruction algorithm based on block compressed sensing.

A direct way to approach reconstruction would be to use sparse recovery on each track separately. However, this leads to blocking artifacts. An alternative solution for block compressed sensing is to reformulate Equation (5) as

$$\underbrace{\begin{bmatrix} Y^{(1)} \\ \vdots \\ Y^{(n_\tau)} \end{bmatrix}}_Y = \underbrace{\begin{bmatrix} \Phi^{(1)} & & \\ & \ddots & \\ & & \Phi^{(n_\tau)} \end{bmatrix}}_\Phi \underbrace{\begin{bmatrix} X^{(1)} \\ \vdots \\ X^{(n_\tau)} \end{bmatrix}}_X \quad (12)$$

Sparse recovery can now be used to solve for all of X from the measurements Y . Though X can now be reconstructed globally, it is often simpler and more efficient to reconstruct spatial slices $X_{\cdot,\ell}$ for the different wavelength channels $\ell \in \{1, \dots, n_c\}$ separately. Note that $X_{\cdot,\ell}$ denotes the ℓ th column of X (equivalent MATLAB-like notation would be $X(:, \ell)$). This yields

$$Y_{\cdot,\ell} = \Phi X_{\cdot,\ell} \quad \text{for each channel } \ell \quad (13)$$

where $Y_{\cdot,\ell}$ —the ℓ th column of Y —are the measurements at the ℓ th wavelength channel. The reconstruction algorithm we employ is the 2D total variation (TV)-minimization (used on each slice separately):

$$\widehat{X}_{\cdot,\ell} = \arg \min_{X_{\cdot,\ell}} \text{TV}_{2D}(X_{\cdot,\ell}) \quad \text{subject to } Y_{\cdot,\ell} = \Phi X_{\cdot,\ell} \quad (14)$$

2.2.3. Multitrack Adaptive Compressed Sensing

This section describes the second of our proposed methods. It involves implementing an adaptive basis scan algorithm where multiple basis coefficients are sampled in parallel.

Preliminary results were published in [43]. Similar to most adaptive basis scans, we use a 2D wavelet basis—specifically, 2D Haar wavelets [37]. Besides their simplicity, Haar wavelets have a property that is critical to our parallelized sampling: when viewed as a n_h -by- n_w pixel 2D pattern, the support (i.e., region of non-zero pixels) of a Haar wavelet is localized and can be completely contained inside a track or subpattern. This allows for sampling a wavelet coefficient using only a subpattern instead of a full n_h -by- n_w pattern. The other subpatterns are free to parallelly sample other wavelet coefficients. Details of the process are described below.

Measurement of Wavelet Coefficients

This problem consists of designing *compound patterns* made up of the subpatterns, such that each of the recorded spectra provide information on the wavelet coefficients. Consider a 2D Haar wavelet $\psi_q \in \mathbb{R}^{n_p}$ labeled by an index q . This, when viewed as a n_h -by- n_w 2D wavelet pattern, can also be divided into n_τ tracks of size n_h -by- (n_w/n_τ) , where the division into tracks can again be described by breaking down ψ_q into submatrices:

$$\psi_q = \begin{bmatrix} \psi_q^{(1)} \\ \psi_q^{(2)} \\ \vdots \\ \psi_q^{(n_\tau)} \end{bmatrix} \quad (15)$$

Here, $\psi_q^{(i)} \in \mathbb{R}^{(n_p/n_\tau)}$ is the i th track of the wavelet. Now, if ψ_q is a Haar wavelet pattern whose support lies entirely in the i th track, then its pixel values must be zero in all other tracks. This would mean that for any track i' other than i , we have $\psi_q^{(i')} = \mathbf{0}$, and ψ_q reduces to:

$$\psi_q = \begin{bmatrix} \mathbf{0} \\ \vdots \\ \psi_q^{(i)} \\ \vdots \\ \mathbf{0} \end{bmatrix} \quad (16)$$

Hence, the sub-matrix (or subpattern) $\psi_q^{(i)}$ can play the role of the full wavelet ψ_q , despite being smaller in size. Importantly, it can still sample dot products:

$$\psi_q^T X = \begin{bmatrix} \mathbf{0}^T & \dots & (\psi_q^{(i)})^T & \dots & \mathbf{0}^T \end{bmatrix} \begin{bmatrix} X^{(1)} \\ \vdots \\ X^{(i)} \\ \vdots \\ X^{(n_\tau)} \end{bmatrix} = (\psi_q^{(i)})^T X^{(i)} \quad (17)$$

Note that the left-hand side is just the wavelet coefficients of the slices of X : $\psi_q^T X = [\psi_q^T X_{\cdot,1}, \dots, \psi_q^T X_{\cdot,n_c}] = [W_{q,1}, \dots, W_{q,n_c}]$. Thus, wavelet subpatterns can be used instead of full wavelets to obtain the wavelet coefficients. This, together with the aforementioned facts that (1) several DMD subpatterns $\zeta^{(i)}$ can be formed parallelly on the DMD and (2) the corresponding spectra can be measured, enables the extraction of multiple wavelet coefficients per slice in a shot.

The precise steps involved in this process are as follows. Let $W_{q_1,\ell}, W_{q_2,\ell}, \dots, W_{q_{n_\tau},\ell}$, for all channels $\ell \in \{1, \dots, n_c\}$, be wavelet coefficients that are to be extracted simultaneously in a shot. To enable this, we require that the supports of the corresponding patterns $\psi_{q_1}, \psi_{q_2}, \dots, \psi_{q_{n_\tau}}$ lie in separate tracks. Without the loss of generality, suppose the support of ψ_{q_i} lies in the i th track for each $i \in \{1, \dots, n_\tau\}$. Then, from (17),

$$[W_{q_{i,1}}, \dots, W_{q_{i,n_c}}] = \psi_{q_i}^T X = \left(\psi_{q_i}^{(i)} \right)^T X^{(i)} \quad (18)$$

Using this equation, we can obtain a formula similar to Equation (11) that extracts the wavelet coefficients from the measured raw spectra. Analogous to Section 2.2.2, we employ the following scaling and shifting operations on the wavelet subpatterns to account for the non-negative and discretized range of the DMD pixel values:

$$\zeta_{q_i}^{(i)} = \frac{2^{b-1} - 1}{2^b - 1} \left(\frac{\psi_{q_i}^{(i)}}{\|\psi_{q_i}^{(i)}\|_\infty} + \mathbf{1}_{n_p}^{(i)} \right) \quad \text{for all tracks } i \quad (19)$$

where b is the number of bits again, and $\mathbf{1}_{n_p}^{(i)}$ is a subpattern $[1, \dots, 1]^T \in \mathbb{R}^{(n_p/n_\tau)}$ of the all-ones pattern $\mathbf{1}_{n_p}$. Again, an additional set of offset subpatterns is required to account for the scaling and shifting mentioned above:

$$\zeta_{\text{offset}}^{(i)} = \frac{2^{b-1} - 1}{2^b - 1} \cdot \mathbf{1}_{n_p}^{(i)} \quad \text{for all tracks } i \quad (20)$$

Now, the final subpatterns $\zeta_{q_i}^{(i)}$ for $i \in \{1, \dots, n_\tau\}$ can be formed parallelly as a compound pattern on the DMD (Figure 1b). The pixels on the i th track of the DMD form the i th subpattern. With the integration time Δt , the recorded spectra (see Equation (4)) are:

$$G(q_1, \dots, q_{n_\tau}) = \begin{bmatrix} \left(\zeta_{q_1}^{(1)} \right)^T X^{(1)} \Delta t \\ \vdots \\ \left(\zeta_{q_{n_\tau}}^{(n_\tau)} \right)^T X^{(n_\tau)} \Delta t \end{bmatrix} \quad (21)$$

A set of measurements from the offset subpatterns is also needed:

$$G_{\text{offset}} = \begin{bmatrix} \left(\zeta_{\text{offset}}^{(1)} \right)^T X^{(1)} \Delta t \\ \vdots \\ \left(\zeta_{\text{offset}}^{(n_\tau)} \right)^T X^{(n_\tau)} \Delta t \end{bmatrix} \quad (22)$$

Denoting the (i, ℓ) th entries of $G(q_1, \dots, q_{n_\tau})$ and G_{offset} by $[G(q_1, \dots, q_{n_\tau})]_{i,\ell}$ and $[G_{\text{offset}}]_{i,\ell}$, respectively, we finally obtain as a result of Equation (18):

$$W_{q_i,\ell} = \frac{2^b - 1}{2^{b-1} - 1} \frac{\|\psi_{q_i}\|_\infty}{\Delta t} ([G(q_1, \dots, q_{n_\tau})]_{i,\ell} - [G_{\text{offset}}]_{i,\ell}) \quad \text{for all } i, \ell \quad (23)$$

Extension of the Adaptive Basis Scan Algorithm to Incorporate Multitrack Measurements

Steps 1–4 in Figure 2 are closely based on Rousset et al. [29,39] and will be described only briefly here. Steps 5 and 6 are novel and designed to incorporate the multitrack requirement into the adaptive algorithm.

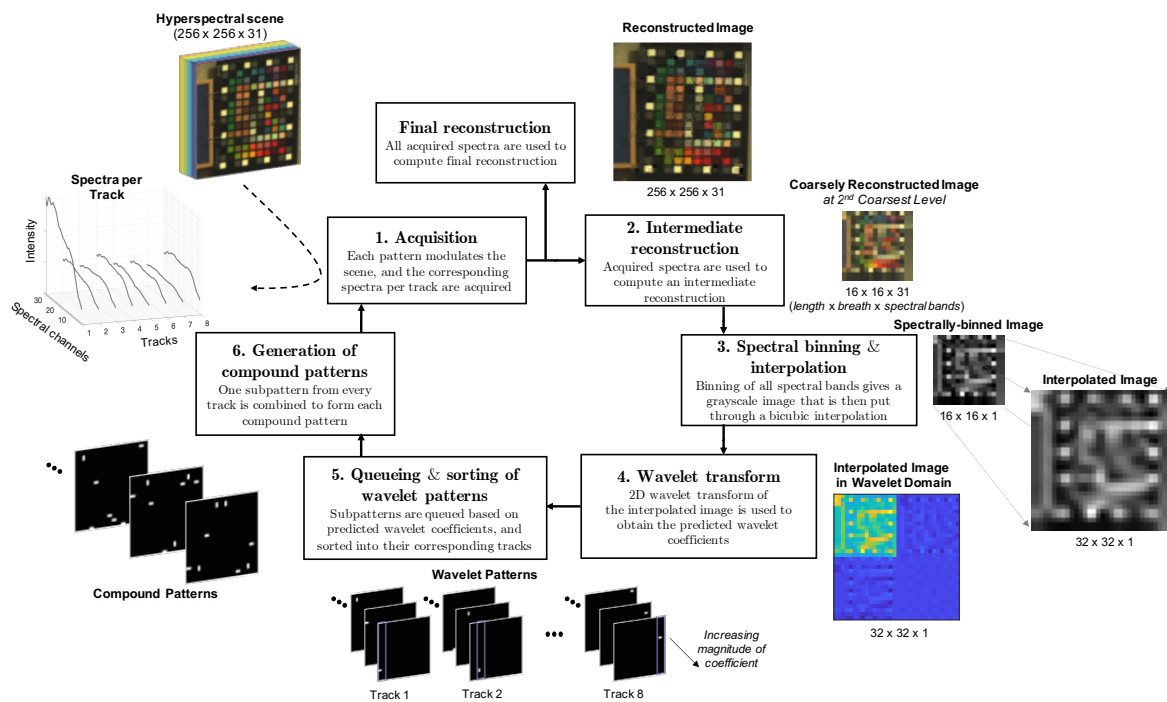


Figure 2. A flow chart of the multitrack adaptive CS scheme for hyperspectral imaging. Steps 1–6 are described briefly here.

The setup is similar to [29,39]. The decomposition level of the wavelet transform is set to some $J \in \mathbb{N}$, which determines the coarsest scale at which information from the scene is captured. However, the need for multitrack sampling imposes a restriction on J due to the requirement that the support of each wavelet pattern must be situated completely within the boundaries of a track. Since the support of a Haar wavelet under decomposition level J can be up to 2^J -by- 2^J pixels in size, we need 2^J to be less than the width of a track in pixels: $2^J \leq n_w/n_\tau$.

Next, the sampling percentages are set for each scale p_J, \dots, p_1 , with $j = J$ being the coarsest, and $j = 1$ being the finest.

The algorithm is divided into $J + 1$ acquisition cycles where wavelet coefficients are measured, and J prediction cycles where the significance of yet-to-be-measured wavelet coefficients is predicted using the already-measured coefficients. The resulting predicted wavelet coefficients then guide the next acquisition cycle and so on.

The algorithm starts off with an acquisition cycle (**Step 1** in Figure 2) where *all* the *approximation wavelet coefficients* of all slices $X_{,\ell}$ of the scene are measured. Unlike [29,39], these are now measured parallelly with the multitrack system using the corresponding approximation compound patterns made up of subpatterns that extract approximation coefficients. The first prediction cycle (described in the next paragraph) predicts the *detail wavelet coefficients* at scale $j = J$, and this is followed by an acquisition cycle that measures detail coefficients at scale $j = J$. The scale is then lowered by one, $j \leftarrow j - 1$, and the cycles repeat.

In a prediction cycle for detail coefficients at scale $j \in \{1, \dots, J\}$, the steps are as follows. First, a low-resolution intermediate reconstruction is obtained by inverse transforming all the wavelet coefficients that were measured or set to zero in all the previous acquisition cycles for every slice (**Step 2** in Figure 2). Binning of all slices at the various wavelength channels gives a grayscale image, which is then oversampled by a factor of 4 in the number of pixels via a bicubic interpolation. (**Step 3** in Figure 2). The oversampled image is then one-level 2D wavelet transformed to give predicted detail wavelet coefficients at scale j (**Step 4** in Figure 2). Until now, all the prediction steps have been similar to the ABS-WP algorithm from [29,39]. The next two steps contain the differences that are needed

to make ABS-WP compatible with the multitrack measurement (see also **Steps 5** and **6** in Figure 2).

Step 5: Predicted wavelet coefficients are queued according to their magnitudes. The higher the magnitude of a predicted wavelet coefficient, the earlier in the queue it appears, as it is expected to be more significant. Each predicted wavelet coefficient will correspond to a unique wavelet, creating an ordered queue of wavelets. Viewing the wavelets as 2D patterns, the support of each wavelet pattern is completely situated within the boundaries of a track, and the corresponding track that the support sits within can be referred to as the pattern's supporting track. The queued wavelet patterns are further sorted into sub-queues based on their supporting tracks while maintaining their ordering within each sub-queue. Based on p_j , a corresponding percentage of the more significant wavelet patterns is retained from each of the sub-queues, while the less significant wavelet patterns (at the back of the queues) are discarded.

However, this sub-queuing process can result in some imaging accuracy losses. In the single-track ABS-WP, the patterns to be used correspond to the p_j most significant predicted coefficients overall. This is different, in general, however, from the p_j most significant coefficients from each sub-queue because some sub-queues might have more of the significant coefficients than other sub-queues.

Step 6: One wavelet pattern from each sub-queue is combined to form a compound pattern: if the wavelet patterns $\psi_{q_1}, \psi_{q_2}, \dots, \psi_{q_{n_\tau}}$ appear first in sub-queues $1, 2, \dots, n_\tau$, respectively, (so that the supporting track of ψ_{q_i} is the i th track), then the first compound pattern is formed by corresponding subpatterns $\zeta_{q_i}^{(i)}$, for $i \in \{1, \dots, n_\tau\}$, obtained from Equation (19). This is continued sequentially, with the second wavelet pattern from each sub-queue combined into the second compound pattern and so on.

This prediction cycle at scale j is followed by an acquisition cycle, as mentioned before, for the measurement of detail coefficients at scale j for all slices $X_{,\ell}$ of the scene. The compound patterns from **Step 6** are used to parallelly measure the detail coefficients that were predicted to be significant in the preceding prediction cycle. The rest of the coefficients at scale j are set to zero. Finally, j is lowered by one and the cycles repeat. The algorithm terminates when $j = 0$.

The final reconstruction is obtained by an inverse wavelet transform, for each slice, of all the wavelet coefficients that were measured (or set to zero) from that slice of the scene in all the acquisition cycles. This is identical to the reconstruction step in [39].

3. Results and Discussion

3.1. Numerical Experiments

All numerical experiments/simulations in this work are conducted using MATLAB 2018a on a HP Z240 Tower Workstation with 4 cores of Intel® Core™ i5-6500 CPU @ 3.20 GHz and 16 GB of RAM.

The performance and applicability of multitrack CS for HSI is assessed computationally on hyperspectral images. The measurements are simulated and fed into the adaptive sampling or reconstruction algorithms being evaluated.

More precisely, a ground truth hyperspectral datacube is treated as the scene to be imaged X . Simulated measurements are computed according to the forward model from Equation (4), where the corresponding subpatterns $\zeta_k^{(i)}$ are generated according to the reconstruction algorithms to be used. For multitrack non-adaptive CS, the subpatterns are obtained from Equation (7), whereas for multitrack adaptive CS, the ABS-WP algorithm and Equation (19) determine the subpatterns. These computations yield the simulated measurements G (see Equations (9) and (21)), which are used to reconstruct the hyperspectral image. The reconstructions are then compared against the ground truth images; the details of these comparisons are provided later in this section.

First, noiseless measurement simulations and reconstruction are performed on three different hyperspectral images: Color Checker, Green Peppers, and U2OS cell, obtained from [44,45]. The former two are 512-by-512-by-31 voxel macroscopic images; the latter

is a 256-by-256-by-231 voxel coherent anti-Stokes Raman Scattering (CARS) hyperspectral image. Both multitrack non-adaptive CS (Section 2.2.2) and multitrack adaptive CS (Section 2.2.3) are compared against benchmarks:

Multitrack full sampling: This involves sampling the whole datacube using a scheme similar to pushbroom. However, for a fair comparison with the multitrack CS methods, it is equipped with the same acquisition architecture as the multitrack methods—the same number of detector tracks, in particular. All spectra are measured from each track of the scene in a ‘multitrack point-scan’ fashion. Hence, no reconstruction step is needed.

Single-track adaptive CS: This involves the sampling of a single spectrum per shot by a SPC-style hyperspectral imager [15], using the adaptive ABS-WP algorithm of [29]. Note that we use an adaptive algorithm for benchmarking here because it generally outperforms single-track non-adaptive CS in terms of speed as well as accuracy.

Details of the compared methods are summarized in Table 1. For multitrack non-adaptive CS, the sparse recovery algorithm used is total variation (TV) minimization with equality constraint, implemented using the TVAL3 package [46]. The settings used are ‘anisotropic TV’ with ‘positivity’ since these tend to give the best results. For multitrack adaptive CS as well as single-track adaptive CS, the decomposition level used is $J = 4$ for Color Checker and Green Peppers, and $J = 3$ for U2OS cell. Higher values of J enable the use of smaller sampling ratios and are generally preferable. However, since the U2OS cell scene is of a smaller pixel count, our use of multitrack adaptive CS requires J to be less than four to ensure that the supports of the wavelet patterns lie completely inside individual tracks.

Table 1. A summary of methods and a comparison of their key features.

	Multitrack Full Sampling	Single-Track Adaptive CS	Multitrack Non-Adaptive CS	Multitrack Adaptive CS
Tracks:	32 tracks	1 track	32 tracks	32 tracks
Sampling ratio:	100%	About 6.2%	About 6.2%	About 6.2%
Integration time:	0.1 s per shot	0.1 s per shot	0.1 s per shot	0.1 s per shot
DMD pattern switch rate:	10,638 frames/s (1-bit binary patterns)	255 frames/s (8-bit grayscale patterns)	255 frames/s (8-bit grayscale patterns)	255 frames/s (8-bit grayscale patterns)

The results are reported in Section 3.2. The performance of the compared methods is quantified by the following metrics:

Reconstruction accuracy: This is quantified by relative error and peak signal-to-noise ratio (PSNR).

Acquisition time: This is composed of (i) the time required to adaptively determine the DMD patterns (0 for non-adaptive), (ii) the time required to load the DMD patterns into a computer’s memory, (iii) the time required to switch the DMD patterns, as per the pattern switch rate, and (iv) the total spectrometer integration time, which is the total number of shots times Δt .

Reconstruction time: This is the time required to compute the final reconstruction.

Total time: Acquisition time + Reconstruction time.

In Section 3.3, we report the simulated effects of the illumination level and noise on imaging accuracy and assess the robustness of the methods developed here. We model measurement noise as Poissonian shot noise together with dark noise. Given a subpattern $\zeta^{(i)}$ for track i and integration time Δt , the recorded noisy spectrum from the track is modeled as

$$\text{Pois}\left(\left(\zeta^{(i)}\right)^T X^{(i)} \Delta t + \alpha \Delta t\right) \quad (24)$$

where $\alpha \in \mathbb{R}_+$ is the dark current in the detector pixels in units of photons $\text{pixel}^{-1} \text{s}^{-1}$, and $\text{Pois}(\mu)$ stands for a Poisson random variable with mean μ .

In low-light imaging, where the scene intensity is low, both shot noise and dark noise become more prominent.

Shot noise: The standard deviation of a $\text{Pois}(\mu)$ random variable is $\sqrt{\mu}$; as a result, the ratio (coefficient of variation) $\frac{\sqrt{\mu}}{\mu}$ grows larger as μ shrinks. The implication is that when the intensity of the scene X is small, the standard deviation of the measured signal is large with respect to its mean.

Dark noise: The $\alpha\Delta t$ term can overpower the $(\zeta^{(i)})^T X^{(i)}\Delta t$ term when the scene intensity is small.

3.2. Simulation Results on Various Hyperspectral Images

The simulation results for the Color Checker and U2OS cell images under noiseless measurement are presented in Figure 3. We evaluate the two proposed multitrack methods against multitrack full sampling and single-track adaptive CS. First, we observe that single-track acquisition is slow: its acquisition time (and total time) is about twice that of multitrack full sampling. Since only one spectrum is measured per shot, a large number of shots are required even with a low sampling ratio of 6.2%. The other three methods, on the other hand, need fewer shots due to their multitrack acquisition architecture.

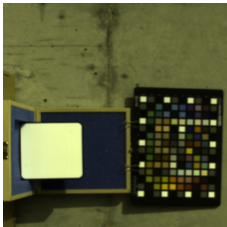
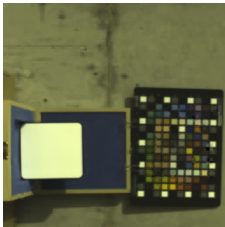
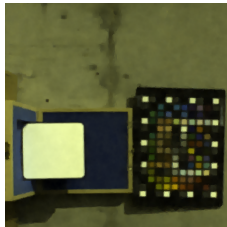
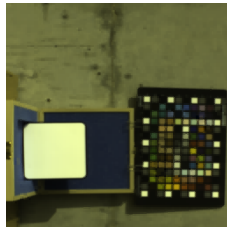
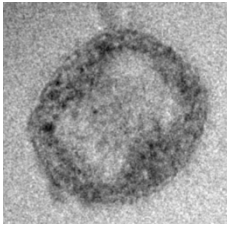
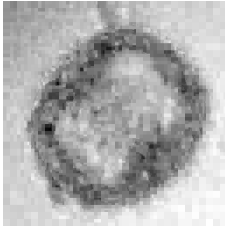
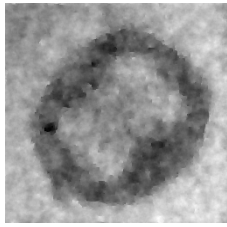
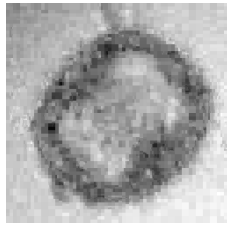
	Multitrack full sampling (ground truth)	Single-track adaptive CS	Proposed: Multitrack non-adaptive CS	Proposed: Multitrack adaptive CS
Color Checker				
	Relative error: 0% PSNR: ∞ Acquisition time: 834 s Reconstruction time: 0 s Total time: 834 s	Relative error: 3.67% PSNR: 37.8 dB Acquisition time: 1695 s Reconstruction time: 1.50 s Total time: 1697 s	Relative error: 4.84% PSNR: 35.4 dB Acquisition time: 54.4 s Reconstruction time: 947 s Total time: 1001 s	Relative error: 3.97% PSNR: 37.1 dB Acquisition time: 62.6 s Reconstruction time: 1.56 s Total time: 64.2 s
U2OS cell				
	Relative error: 0% PSNR: ∞ Acquisition time: 206 s Reconstruction time: 0 s Total time: 206 s	Relative error: 2.06% PSNR: 39.5 dB Acquisition time: 429 s Reconstruction time: 4.37 s Total time: 433 s	Relative error: 3.11% PSNR: 35.9 dB Acquisition time: 14.5 s Reconstruction time: 478 s Total time: 493 s	Relative error: 2.10% PSNR: 39.3 dB Acquisition time: 16.4 s Reconstruction time: 4.38 s Total time: 20.8 s

Figure 3. Simulations are carried out on the hyperspectral scenes Color Checker and U2OS cell. The reconstruction results are presented for each of the methods tested. For the Color Checker scene, pseudo-RGB versions of the reconstructed images are displayed in the table. For the U2OS cell scene, the 2964.5 cm^{-1} band of the reconstructed image is displayed.

Multitrack non-adaptive CS had the shortest acquisition time—about 6–7% of the acquisition time of full sampling enabled by the employed sampling ratio of 6.2%. For the 512-by-512-by-31 voxel Color Checker scene, full sampling required 8192 shots, whereas sampling at 6.2% amounted to 516 shots. Note that the 516 shots here include 10 extra shots corresponding to the offset patterns mentioned in Section 2.2. A similar comparison applies

to the U2OS cell scene, which is 256-by-256-by-231 voxels. A low sampling ratio, together with multitrack acquisition, reduced the acquisition time significantly. Furthermore, no adaptive prediction steps were needed during the acquisition process, unlike for multitrack adaptive CS. Hence, the pattern sequence is not computed on the fly and can be pre-programmed into the DMD's field-programmable gate array (FPGA) for increased speed. However, due to long reconstruction times (947 s for the Color Checker scene and 478 s for the U2OS cell scene), the total time for multitrack non-adaptive CS is even longer than that of full sampling. Thus, despite having a fast acquisition step, the long reconstruction time necessitates offline post-processing.

In terms of total time, multitrack adaptive CS performs the best. It is faster than multitrack non-adaptive CS due to a much shorter reconstruction time, even though the acquisition time is up to 8 s higher due to the adaptive steps. Multitrack adaptive CS is also faster than full sampling, again, because of the small sampling ratio that required fewer shots.

For the multitrack methods, an increase in the number of tracks improves the imaging speed due to a higher extent of parallelization in sampling. However, this is valid only up to a limit in the case of multitrack adaptive CS—the tracks still have to be wide enough to contain the supports of the wavelet patterns, as was discussed earlier.

Regarding reconstruction accuracy, the simulation results show that multitrack adaptive CS outperforms its non-adaptive counterpart for both the Color Checker and the U2OS cell scene; the PSNR for multitrack adaptive CS is 2–3 dB higher. Hence, there is evidence that the wavelet coefficients measured capture the important information from the scene effectively. This discrepancy in accuracy could also be due to the scenes being more easily representable in the wavelet basis or due to the scenes not having sufficiently low total variation as would be necessary for TV-based sparse recovery. Full sampling achieves the best imaging accuracy; however, the (relative) errors are smaller than 5% for the multitrack CS methods. The slight loss of accuracy for a huge gain in speed presents a good tradeoff in favor of multitrack adaptive CS in particular.

Besides the above comparisons, one needs to make sure that the accuracy of multitrack adaptive CS is not much behind that of single-track adaptive CS. The simulation results indeed confirm that despite the losses induced by the sub-queue formation in multitrack adaptive CS, the PSNR discrepancy of the two methods does not exceed 1 dB. Hence, this shows that the ABS-WP algorithm has been successfully extended to the multitrack measurement scenario.

Next, Figure 4 shows a similar comparison between the four methods for the Green Peppers image. The first row of Figure 4 yields similar conclusions as Figure 3: multitrack non-adaptive CS has the shortest acquisition time; multitrack adaptive CS has the shortest total time; and the relative errors of the multitrack CS methods, though higher than in Figure 3, are still only about 6–7%. In the second, third and fourth rows of Figure 4, we demonstrate the potential applicability of multitrack CS HSI to agricultural monitoring. Reconstructed spectra from each of the compared methods are plotted in the second row. The dotted lines are from a point on the green leaves background; the dashed lines are from a point on the pepper near the top left corner; and the solid lines are from a point on a chili at the bottom right corner, as indicated on the ground truth image. The multitrack CS methods can distinguish the three, as can full sampling and single-track. The third row shows the errors of the reconstructed spectra from the various methods relative to the ground truth spectra. The errors are low for the multitrack CS methods and do not prevent us from distinguishing the different regions while still being fast. The exact trends of the error curves likely depend on the shape of the spectra and the intensity levels. Finally, the fourth row displays the pixel-wise ratio of the spectral bands at 560 and 680 nm. The ratio images clearly distinguish the peppers from the background for each of the compared methods, possibly making it easier to perform semantic segmentation. Note that the red regions at the bottom right corners correspond to chilies.

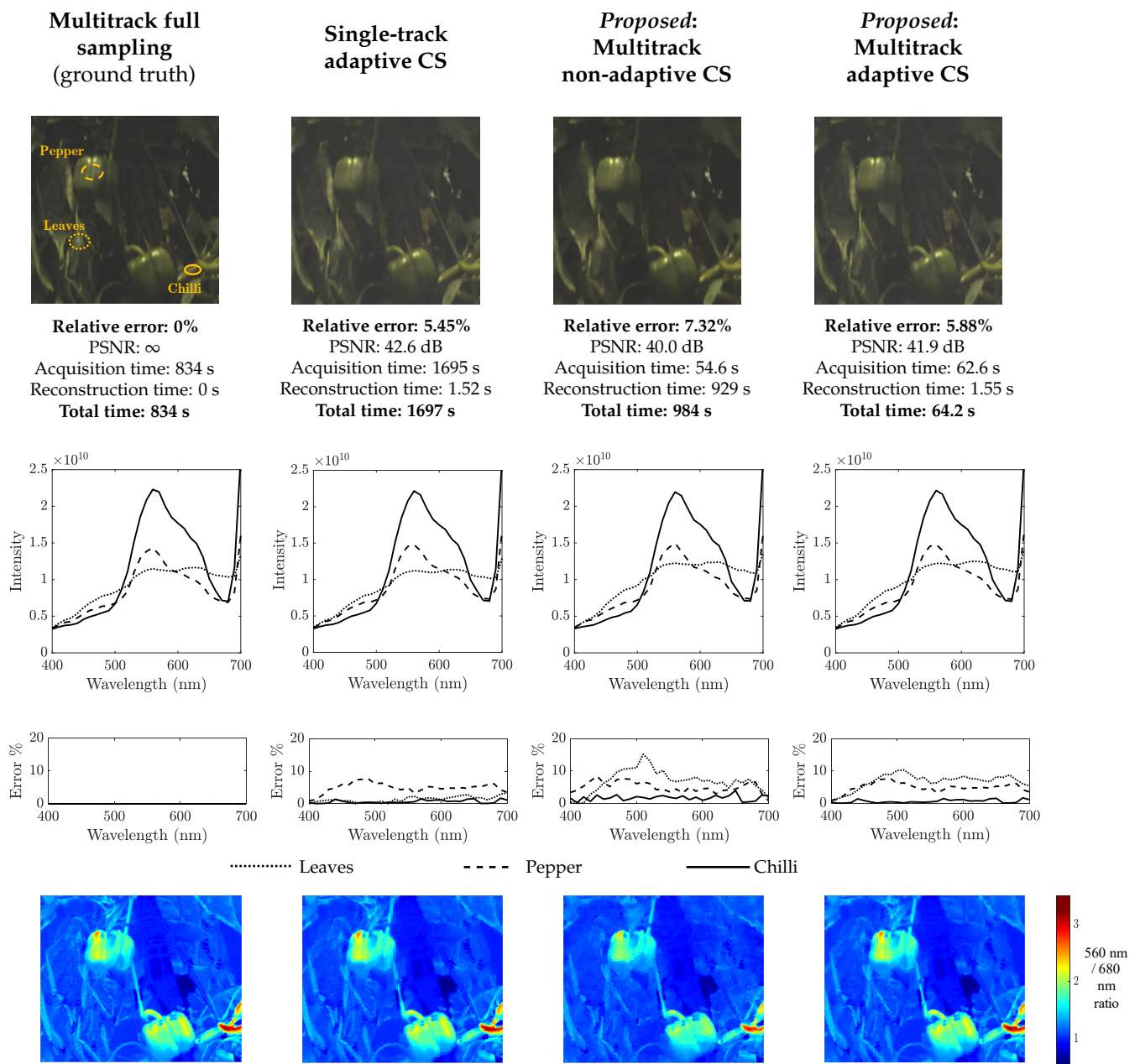


Figure 4. Simulations are carried out on the hyperspectral scene Green Peppers. The reconstruction results are presented for each of the methods tested. Pseudo-RGB versions of the reconstructed images are displayed in the first row. Reconstructed spectra are plotted in the second row. The dotted lines (.....) are from a point on the green leaves background; the dashed lines (- - - -) are from a point on the pepper near the top left corner; and the solid lines (—) are from a point on a chilli at the bottom right corner, as indicated on the ground truth image. The third row shows the errors of the reconstructed spectra relative to the ground truth spectra. Finally, the fourth row displays the pixel-wise ratio of the spectral bands at 560 and 680 nm.

3.3. The Effect of Illumination Level and Noise

We modeled measurement noise as Poissonian shot noise with dark noise, as described in Section 3.1. The factors determining the severity of noise were intensity of the scene X , integration time Δt and dark current α . The effects of these factors on the CS reconstructions of the U2OS cell scene are examined here via simulations for the multitrack methods. We do not analyze single-track CS further since we established in Section that it is much slower than the multitrack methods.

First, we set $\alpha = 0$ and note that X and Δt play a similar role in the resulting measurement:

$$\text{Pois}\left(\left(\tilde{\zeta}^{(i)}\right)^T X^{(i)} \Delta t\right)$$

Hence, it is reasonable to evaluate the effect of the product $X\Delta t$ on the reconstruction accuracy instead of using X and Δt individually. Furthermore, $X\Delta t$ has units of photons and, therefore, is physically meaningful as the photon counts emanating from the scene during the integration time interval. As a result, we quantify the illumination level by $\max(X\Delta t)$ —the max photon count per voxel emanating from the scene during the time interval Δt .

Figure 5 shows the simulation results—the effect of illumination level on the reconstructed image accuracy (in PSNR)—for multitrack full sampling, multitrack non-adaptive CS and multitrack adaptive CS. The illumination level plotted on the horizontal axis decreases from left to right, corresponding to an increasing prominence of shot noise.

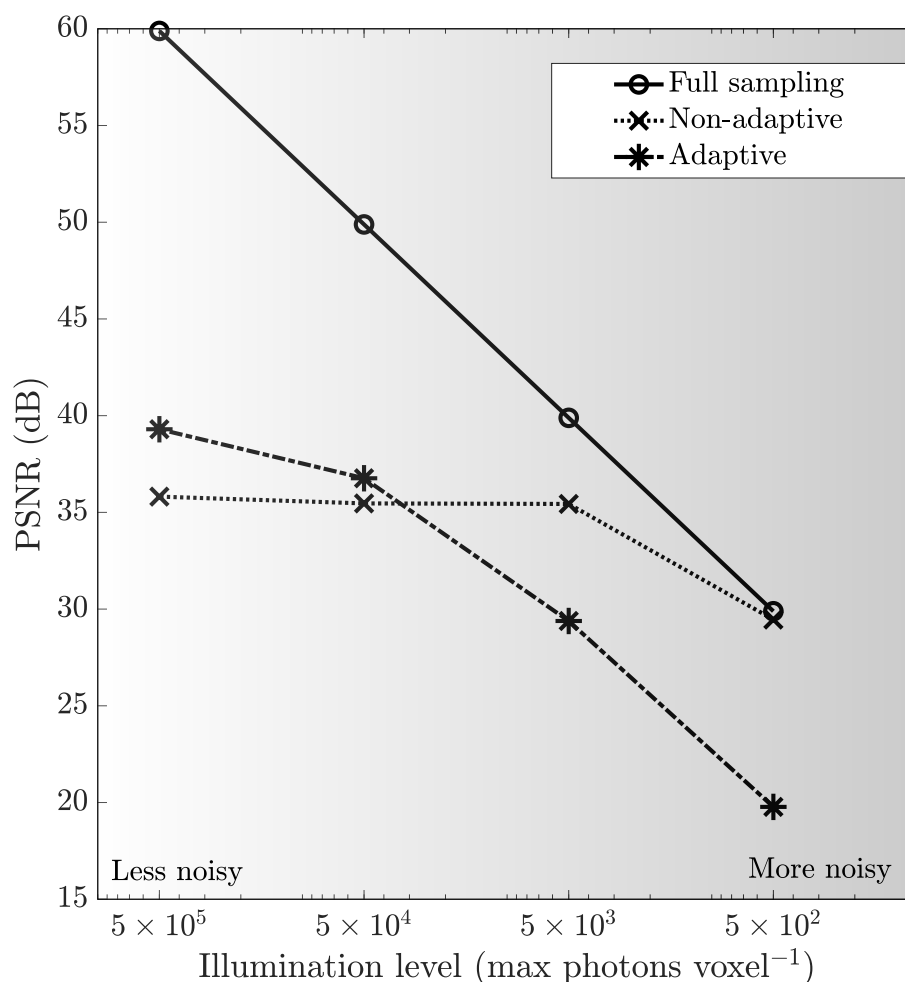


Figure 5. The effect of the illumination level on the reconstructed image PSNR of the U2OS cell scene for multitrack full sampling, multitrack non-adaptive CS and multitrack adaptive CS. The illumination level (horizontal axis) decreases from left to right, corresponding to an increasing prominence of shot noise. The gray shading illustrates the illumination level. The dark current α is fixed at 0.

The reconstruction accuracy is highest for full sampling, similar to Section 3.2, across all illumination levels. However, the accuracy of non-adaptive CS starts approaching that of full sampling at lower illumination levels of $\sim 5 \times 10^2$ photons voxel⁻¹. We also note

that for illumination levels higher than $\sim 5 \times 10^4$ photons voxel $^{-1}$, the accuracy of adaptive CS overtakes that of non-adaptive CS.

A prominent trend is that the reconstruction accuracies of full sampling and adaptive CS are more sensitive to the illumination level as compared to non-adaptive CS. The DMD patterns used in the former two have small supports, such that only a small percentage of incident light is reflected towards the detector. This leads to a weaker signal at the detector, which is more susceptible to shot noise. On the contrary, the DMD patterns used in non-adaptive CS reflect about 50% of the incident light towards the detector, which is much higher than the other two methods. The resultant stronger signal at the detector is thus less affected by shot noise.

Noting that this sensitivity is a measure of robustness to shot noise, we can conclude from comparing the slopes of the PSNR-illumination plots that the robustness of our CS methods is better than the robustness of full sampling.

Next, we evaluate the effect of the dark current by varying α . We fix the max scene intensity at 5×10^4 photons s $^{-1}$ voxel $^{-1}$ and the integration time 0.1 s, yielding an illumination level of 5×10^3 photons voxel $^{-1}$. Figure 6 shows the simulation results—the reconstruction accuracy across a wide range of dark current values. The plots indicate that full sampling is much more sensitive to dark current as compared to adaptive and non-adaptive CS. This gives the CS methods an advantage—their robustness to dark noise can allow the use of cheaper detectors or relax the cooling requirements of detectors.

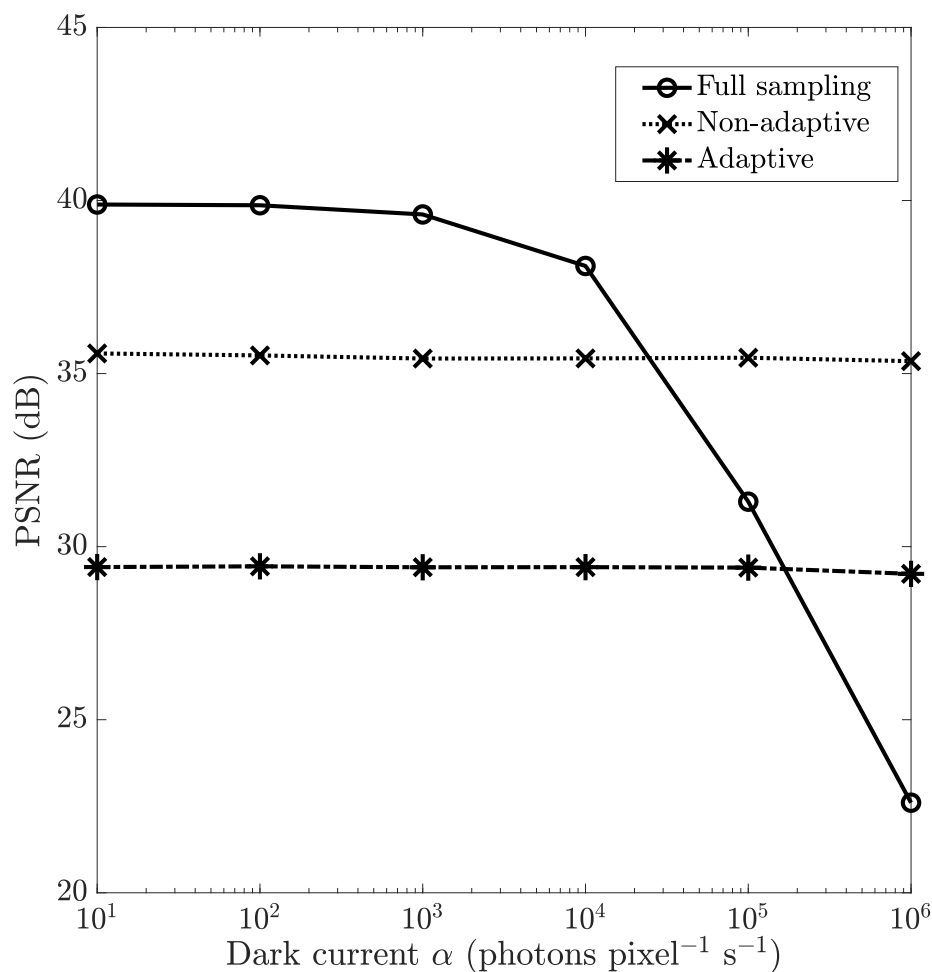


Figure 6. The effect of dark current α on the reconstructed image PSNR of the U2OS cell scene for multitrack full sampling, multitrack non-adaptive CS and multitrack adaptive CS. The max scene intensity is fixed at 5×10^4 photons s $^{-1}$ voxel $^{-1}$, and the integration time is 0.1 s

These discrepancies could again be attributed to signal strengths at the detector, as mentioned earlier. Full sampling uses delta function subpatterns, which have smaller supports than the wavelet patterns used in adaptive CS and the random patterns used in non-adaptive CS. Furthermore, the CS methods have a regularization method built into them—sparsity in the wavelet domain for adaptive CS, and low TV norms for non-adaptive CS—making them more noise-robust in general.

Lastly, we examine the effect of the integration time. Both reconstruction accuracy and acquisition time are affected by this quantity, leading to tradeoffs. The effect of the integration time on the accuracy was effectively captured in Figure 5 since the illumination level was determined by $X\Delta t$, which is linear in Δt . The Δt –PSNR relationship is plotted explicitly in the top panel of Figure 7 for the multitrack methods. Here, we have fixed $\alpha = 0$ and $\max(X) = 5 \times 10^4$ photons s^{-1} voxel $^{-1}$.

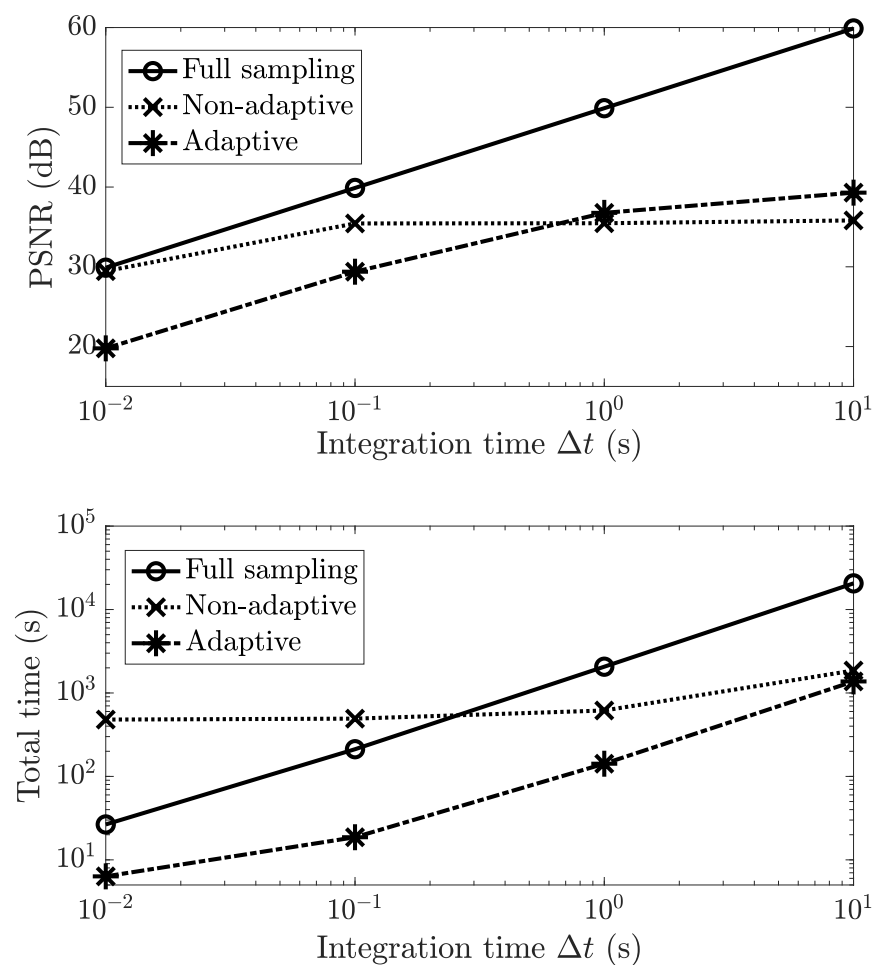


Figure 7. The effect of the integration time Δt on the U2OS cell scene’s reconstruction PSNR and total time (acquisition + reconstruction) for multitrack full sampling, multitrack non-adaptive CS and multitrack adaptive CS. $\alpha = 0$ and $\max(X) = 5 \times 10^4$ photons s^{-1} voxel $^{-1}$ are fixed.

The acquisition time and total time are related to the integration time as per the following equation:

$$\begin{aligned} \text{total time} = & (\text{number of shots} \times \Delta t) \\ & + \text{DMD-pattern-related time} + \text{reconstruction time} \quad (25) \end{aligned}$$

The DMD-pattern-related time is independent of the integration time and comprises of (i) the time required to adaptively determine the DMD patterns (0 s for non-adaptive),

(ii) the time required to load the DMD patterns into a computer's memory and (iii) the time required to switch the DMD patterns, as per the pattern switch rate. Similarly, the reconstruction time is also independent of the integration time. These components are offset terms in Equation (25). The remaining term provides a linear relationship between the total time and Δt , the slope of which is governed by the sampling ratio. The bottom panel of Figure 7 plots these relationships for the three multitrack methods.

The total time curve for full sampling is the steepest due to a larger number of shots. Non-adaptive CS, on the other hand, has the highest reconstruction time, leading to a large offset when $\Delta t \rightarrow 0$. The lowest total time is achieved by adaptive CS, as was also shown in Section 3.2.

Both panels in Figure 7 can be used together for choosing the optimal multitrack method for the given accuracy and time requirements. Suppose a minimum PSNR must be achieved. Then the top panel would yield the integration times required by each of the methods to achieve this PSNR threshold. Based on the integration times, the bottom panel would then predict the total time required by the three methods, allowing us to choose the one with the smallest time. Conversely, if we have a maximum total time requirement, a similar procedure based on Figure 7 could then be used to choose the method that achieves the highest PSNR under this time constraint.

4. Conclusions

In this paper, we have developed two methods for multitrack compressed-sensing-based hyperspectral imaging—a non-adaptive approach based on sparse recovery and an adaptive approach based on sampling significant wavelet coefficients of the hyperspectral scene.

We have verified via simulations that both the methods can achieve short measurement times while maintaining high imaging accuracy. However, they have different advantages: the adaptive method enjoys a much shorter reconstruction time and computational burden, whereas the non-adaptive method is more robust to noise.

Next, we have also shown that the two methods are generalizable to different types of scenes—the reflectance of macroscopic objects as well as Raman signals of cells. This is likely because the methods do not rely upon information from fixed data sets.

Overall, the CS-HSI methods achieve a reconstruction accuracy close to full sampling, and the noise-robustness is even better than full sampling. This, along with the high acquisition speed as well as high reconstruction speed in the case of multitrack adaptive CS, demonstrates these methods to be effective.

Author Contributions: Conceptualization, S.K. and D.Y.; methodology, S.K. and D.Y.; software, S.K.; validation, S.K., E.L., C.Y.T. and D.Y.; formal analysis, S.K., E.L. and D.Y.; investigation, S.K.; resources, D.Y.; data curation, S.K. and D.Y.; writing—original draft preparation, S.K. and D.Y.; writing—review and editing, S.K., E.L., C.Y.T. and D.Y.; visualization, S.K., E.L. and D.Y.; supervision, C.Y.T. and D.Y.; project administration, C.Y.T. and D.Y.; funding acquisition, C.Y.T. and D.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the National Research Foundation, Prime Minister's Office, Singapore, under its Campus for Research Excellence and Technological Enterprise (CREATE) program, through Singapore MIT Alliance for Research and Technology (SMART): Critical Analytics for Manufacturing Personalised-Medicine (CAMP) Inter-Disciplinary Research Group. It is also supported by the Agency for Science Technology and Research (A*STAR), Singapore, through its internship programme; and is co-supported by A*STAR and Nanyang Technological University, Singapore, through its joint Final Year Project.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The hyperspectral image data used in this paper was obtained from the public database <http://icvl.cs.bgu.ac.il/hyperspectral/>, accessed on 20 November 2019. and (upon request) from <http://dx.doi.org/10.17035/d.2015.100098>, accessed on 10 October 2019.

Acknowledgments: The authors would like to express their thanks to Joel Yeo for the discussions on compressed sensing. The authors would also like to thank Masia et al. and Cardiff University for making their Raman hyperspectral image data available to us.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

HSI	Hyperspectral imaging
CS	Compressed sensing
CASSI	Coded aperture snapshot spectral imaging
CS-HSI	Compressed sensing based hyperspectral imaging
SPC	Single pixel camera
DMD	Digital micromirror device
ABS-WP	Adaptive basis scan with wavelet prediction
TV	Total variation
CARS	Coherent anti-Stokes Raman Scattering
PSNR	Peak signal-to-noise ratio
RGB	Red Green Blue
FPGA	Field-programmable gate array

References

- Lu, G.; Fei, B. Medical hyperspectral imaging: A review. *J. Biomed. Opt.* **2014**, *19*, 010901. [[CrossRef](#)]
- Li, Q.; Wang, W.; Ma, C.; Zhu, Z. Detection of physical defects in solar cells by hyperspectral imaging technology. *Opt. Laser Technol.* **2010**, *42*, 1010–1013. [[CrossRef](#)]
- Dale, L.M.; Thewis, A.; Boudry, C.; Rotar, I.; Dardenne, P.; Baeten, V.; Pierna, J.A.F. Hyperspectral Imaging Applications in Agriculture and Agro-Food Product Quality and Safety Control: A Review. *Appl. Spectrosc. Rev.* **2013**, *48*, 142–159. [[CrossRef](#)]
- Goetz, A.F. Three decades of hyperspectral remote sensing of the Earth: A personal view. *Remote Sens. Environ.* **2009**, *113*, S5–S16. [[CrossRef](#)]
- Qin, J.; Chao, K.; Kim, M. Chapter 6—Introduction to Raman Chemical Imaging Technology. In *Computer Vision Technology for Food Quality Evaluation*, 2nd ed.; Sun, D.W., Ed.; Academic Press: Cambridge, MA, USA, 2016; pp. 141–171.
- Fisher, A.C.; Kamga, M.H.; Agarabi, C.; Brorson, K.; Lee, S.L.; Yoon, S. The Current Scientific and Regulatory Landscape in Advancing Integrated Continuous Biopharmaceutical Manufacturing. *Trends Biotechnol.* **2019**, *37*, 253–267. [[CrossRef](#)] [[PubMed](#)]
- Ündey, C.; Ertunç, S.; Mistretta, T.; Looze, B. Applied advanced process analytics in biopharmaceutical manufacturing: Challenges and prospects in real-time monitoring and control. *J. Process Control* **2010**, *20*, 1009–1018. [[CrossRef](#)]
- Hagen, N.A.; Kudenov, M.W. Review of snapshot spectral imaging technologies. *Opt. Eng.* **2013**, *52*, 1–23. [[CrossRef](#)]
- Candes, E.J.; Romberg, J.; Tao, T. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory* **2006**, *52*, 489–509. [[CrossRef](#)]
- Donoho, D.L. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [[CrossRef](#)]
- August, I.Y.; Stern, A. Spectral and Hyperspectral Compressive Imaging. In *Optical Compressive Imaging*; Stern, A., Ed.; CRC Press: Boca Raton, FL, USA, 2017.
- Wagadarikar, A.; John, R.; Willett, R.; Brady, D. Single disperser design for coded aperture snapshot spectral imaging. *Appl. Opt.* **2008**, *47*, B44–B51. [[CrossRef](#)] [[PubMed](#)]
- Arce, G.R.; Brady, D.J.; Carin, L.; Arguello, H.; Kittle, D.S. Compressive Coded Aperture Spectral Imaging: An Introduction. *IEEE Signal Process. Mag.* **2014**, *31*, 105–115. [[CrossRef](#)]
- Cao, X.; Yue, T.; Lin, X.; Lin, S.; Yuan, X.; Dai, Q.; Carin, L.; Brady, D.J. Computational Snapshot Multispectral Cameras: Toward dynamic capture of the spectral world. *IEEE Signal Process. Mag.* **2016**, *33*, 95–108. [[CrossRef](#)]
- Sun, T.; Kelly, K. Compressive sensing hyperspectral imager. In *Computational Optical Sensing and Imaging*; Optical Society of America: Washington, DC, USA, 2009.
- Baek, S.H.; Kim, I.; Gutierrez, D.; Kim, M.H. Compact Single-Shot Hyperspectral Imaging Using a Prism. *ACM Trans. Graph.* **2017**, *36*. [[CrossRef](#)]
- Jeon, D.S.; Baek, S.H.; Yi, S.; Fu, Q.; Dun, X.; Heidrich, W.; Kim, M.H. Compact Snapshot Hyperspectral Imaging with Diffracted Rotation. *ACM Trans. Graph.* **2019**, *38*. [[CrossRef](#)]
- Rani, M.; Dhok, S.B.; Deshmukh, R.B. A Systematic Review of Compressive Sensing: Concepts, Implementations and Applications. *IEEE Access* **2018**, *6*, 4875–4894. [[CrossRef](#)]
- Marques, E.C.; Maciel, N.; Naviner, L.; Cai, H.; Yang, J. A Review of Sparse Recovery Algorithms. *IEEE Access* **2019**, *7*, 1300–1322. [[CrossRef](#)]

20. Aharon, M.; Elad, M.; Bruckstein, A. K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation. *IEEE Trans. Signal Process.* **2006**, *54*, 4311–4322. [[CrossRef](#)]
21. Choi, I.; Jeon, D.S.; Nam, G.; Gutierrez, D.; Kim, M.H. High-quality hyperspectral reconstruction using a spectral prior. *ACM Trans. Graph.* **2017**, *36*, 1–13. [[CrossRef](#)]
22. Mousavi, A.; Patel, A.B.; Baraniuk, R.G. A deep learning approach to structured signal recovery. In Proceedings of the 2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton), Monticello, IL, USA, 29 September–2 October 2015; pp. 1336–1343. [[CrossRef](#)]
23. Li, R.; Zheng, Y.; Wen, D.; Song, Z. A deep learning approach to real-time recovery for compressive hyper spectral imaging. In Proceedings of the 2017 IEEE 3rd Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 3–5 October 2017; pp. 1030–1034. [[CrossRef](#)]
24. Xiong, Z.; Shi, Z.; Li, H.; Wang, L.; Liu, D.; Wu, F. HSCNN: CNN-Based Hyperspectral Image Recovery from Spectrally Undersampled Projections. In Proceedings of the 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), Venice, Italy, 22–29 October 2017; pp. 518–525. [[CrossRef](#)]
25. Wang, L.; Zhang, T.; Fu, Y.; Huang, H. HyperReconNet: Joint Coded Aperture Optimization and Image Reconstruction for Compressive Hyperspectral Imaging. *IEEE Trans. Image Process.* **2019**, *28*, 2257–2270. [[CrossRef](#)]
26. Deutsch, S.; Averbush, A.; Dekel, S. Adaptive Compressed Image Sensing Based on Wavelet Modeling and Direct Sampling. In Proceedings of the International Conference on SAMPLing Theory and Applications (SAMPTA'09), Marseille, France, 18–22 May 2009.
27. Averbuch, A.; Dekel, S.; Deutsch, S. Adaptive Compressed Image Sensing Using Dictionaries. *SIAM J. Imag. Sci.* **2012**, *5*, 57–89. [[CrossRef](#)]
28. Dai, H.; Gu, G.; He, W.; Liao, F.; Zhuang, J.; Liu, X.; Chen, Q. Adaptive compressed sampling based on extended wavelet trees. *Appl. Opt.* **2014**, *53*, 6619–6628. [[CrossRef](#)]
29. Rousset, F.; Ducros, N.; Farina, A.; Valentini, G.; D’Andrea, C.; Peyrin, F. Adaptive Basis Scan by Wavelet Prediction for Single-Pixel Imaging. *IEEE Trans. Comput. Imag.* **2017**, *3*, 36–46. [[CrossRef](#)]
30. Lu, Y.; Xu, X.; Wang, L. Smart manufacturing process and system automation—A critical review of the standards and envisioned scenarios. *J. Manuf. Syst.* **2020**, *56*, 312–325. [[CrossRef](#)]
31. Huang, S.H.; Pan, Y.C. Automated visual inspection in the semiconductor industry: A survey. *Comput. Ind.* **2015**, *66*, 1–10. [[CrossRef](#)]
32. Gehm, M.E.; John, R.; Brady, D.J.; Willett, R.M.; Schulz, T.J. Single-shot compressive spectral imaging with a dual-disperser architecture. *Opt. Express* **2007**, *15*, 14013–14027. [[CrossRef](#)] [[PubMed](#)]
33. Duarte, M.F.; Davenport, M.A.; Takhar, D.; Laska, J.N.; Sun, T.; Kelly, K.F.; Baraniuk, R.G. Single-pixel imaging via compressive sampling. *IEEE Signal Process. Mag.* **2008**, *25*, 83–91. [[CrossRef](#)]
34. Wagadarikar, A.A.; Pitsianis, N.P.; Sun, X.; Brady, D.J. Video rate spectral imaging using a coded aperture snapshot spectral imager. *Opt. Express* **2009**, *17*, 6368–6388. [[CrossRef](#)]
35. Kittle, D.; Choi, K.; Wagadarikar, A.; Brady, D.J. Multiframe image estimation for coded aperture snapshot spectral imagers. *Appl. Opt.* **2010**, *49*, 6824–6833. [[CrossRef](#)] [[PubMed](#)]
36. Magalhaes, F.; Araújo, F.M.; Correia, M.; Abolbashari, M.; Farahi, F. High-resolution hyperspectral single-pixel imaging system based on compressive sensing. *Opt. Eng.* **2012**, *51*, 1–7. [[CrossRef](#)]
37. Mallat, S. *A Wavelet Tour of Signal Processing*, 3rd ed.; Academic Press, Inc.: Cambridge, MA, USA, 2008.
38. Hahn, J.; Debes, C.; Leigsnering, M.; Zoubir, A.M. Compressive sensing and adaptive direct sampling in hyperspectral imaging. *Digit. Signal Process.* **2014**, *26*, 113–126. [[CrossRef](#)]
39. Rousset, F.; Ducros, N.; Peyrin, F.; Valentini, G.; D’Andrea, C.; Farina, A. Time-resolved multispectral imaging based on an adaptive single-pixel camera. *Opt. Express* **2018**, *26*, 10550–10558. [[CrossRef](#)] [[PubMed](#)]
40. Lu, G. Block Compressed Sensing of Natural Images. In Proceedings of the 2007 15th International Conference on Digital Signal Processing, Cardiff, UK, 1–4 July 2007; pp. 403–406. [[CrossRef](#)]
41. Fowler, J.E.; Mun, S.; Tramel, E.W. Block-Based Compressed Sensing of Images and Video. *Found. Trends Signal Process.* **2012**, *4*, 297–416. [[CrossRef](#)]
42. Pfeffer, Y.; Zibulevsky, M. *A Micro-Mirror Array Based System for Compressive Sensing of Hyperspectral Data*; Technical Report; Computer Science Department, Technion: Haifa, Israel, 2010.
43. Kubal, S.; Lee, E.; Tay, C.Y.; Yong, D. Multitrack Adaptive Compressive Hyperspectral Imaging for Cell Monitoring Applications. In *Laser Science*; Optical Society of America: Washington, DC, USA, 2020. [[CrossRef](#)]
44. Arad, B.; Ben-Shahar, O. Sparse Recovery of Hyperspectral Signal from Natural RGB Images. In Proceedings of the Computer Vision (ECCV 2016), Amsterdam, The Netherlands, 11–14 October 2016; pp. 19–34.
45. Masia, F.; Karuna, A.; Borri, P.; Langbein, W. Hyperspectral image analysis for CARS, SRS, and Raman data. *J. Raman Spectrosc.* **2015**, *46*, 727–734. [[CrossRef](#)] [[PubMed](#)]
46. Li, C. An Efficient Algorithm for Total Variation Regularization with Applications to the Single Pixel Camera and Compressive Sensing. Master’s Thesis, Rice University, Houston, TX, USA, 2010.