Hessian Free Convolutional Dictionary Learning for Hyperspectral Imagery With Application to Compressive Chromo-Tomography

XUESONG ZHANG*, (Member, IEEE), BAOPING LI*, (Graduate Student Member, IEEE), AND JING JIANG**, (Member, IEEE)
1School of Computer Science, Beijing University of Posts and Telecommunications, Beijing 100876, China
2Department of Communication Engineering, Beijing Union University, Beijing 100101, China
Corresponding author: Xuesong Zhang (xuesong_zhang@bupt.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61871055, and in part by the Fundamental Research Funds for the New Start Plan Project of Beijing Union University under Grant Zk10201604.

ABSTRACT  Convolutional dictionary learning (CDL) is an unsupervised learning method to seek a translation-invariant sparse representation for signals, and has gained a lot of interest in various image processing and computer vision applications. However, 3D hyperspectral images pose unique challenges due to their high dimensionality and complex structures, making optimization of the dictionary and its application to inverse problems difficult. This paper proposes an efficient CDL algorithm that neither explicit evaluation of the Hessians nor their inversion is required in the optimization process, which leads to substantial acceleration and memory savings. Furthermore, we exploit the learned kernels as the convolutional sparse coding (CSC) image prior for the compressive chromo-tomographic (CCT) reconstruction problem, and examine the usability and performances of the proposed method for CCT reconstruction. Numerical experiments show that, for CCT, 1) the proposed CSC can provide an efficient representation for HSI by using several tens of 3D filters; 2) the learned convolutional dictionary has reliable generalization capability; and 3) the proposed CSC-based method outperforms the classical reconstruction method using an analytic sparsifying basis.

INDEX TERMS Dictionary learning, compressive hyperspectral imaging, alternating direction method of multipliers (ADMM), preconditioning.

I. INTRODUCTION
Hyperspectral imagery (HSI) measures the radiation intensity of every pixel in the scene over many wavelength bands. These spatially registered spectral images exhibit three important attributes: local spatial smoothness, high spectral correlation and nonlocal spatial similarity, which make dictionary learning (DL) a well-suited approach to seeking for a sparse and redundant representation model for various HSI analysis and processing tasks, including hyperspectral classification [1]–[3], hyperspectral image denoising [4], anomaly detection [5], hyperspectral image super-resolution [6]–[8], and hyperspectral compressive sensing [9], [10], among others.

The DL problem probes for a set of prototype signals or features, constituting a usually overcomplete dictionary, from a training database. This data-dependent methodology distinguishes from classical analytic transforms, such as Wavelet [11] and Contourlet [12], in that the resultant dictionary can better adapt to any family of signals via learning whereas analytic transforms are usually designed for some particular class of signals, for example, for piece-wise constant signals and for image edges. The flexibility and adaptability of DL has made it a rather appealing sparse representation approach and its evolution has displayed three important stages that are worthy of mention:

1) Patched-based dictionary learning, extracts the so-called dictionary atoms from overlapped image patches, forming the columns of the dictionary D, and approximates every image patch with a linear combination $x \approx Dz$, where z is
the sparse vector that plays an important role in the unique representation and in the interpretability. This data-driven image modeling has demonstrated much improved performances over generic image models in many practical applications [13]. However, their patch-wise manipulation gives rise to two restrictions on $D$, unstructuredness and shift-variance. The former determines that explicit matrix-vector operations dominate the numerical expense while the latter means that dictionaries may contain many shifted versions of the same atoms. These two issues impose severe computational and memory burdens on the dictionary learning algorithms, such as the method of optimal directions (MOD) [14], the K-SVD [15], and the generalized PCA (GPCA) [16]. A novel PCA-based dictionary learning technique is proposed for multi-modality image fusion, where multiple sub-dictionaries are trained, each for a different cluster of pixels sharing common high-frequency features [17].

- **Double-sparsity dictionary learning**, trains a sparse atom representation matrix $A$ over a preselected dictionary $Φ$, e.g., Discrete Cosine Transform (DCT) or Wavelet transform, and therefore the dictionary takes the form $D = ΦA$ [18]. Some follow-up methods extend this idea to multiscale [19], orthogonal [20], or online [21] dictionary learning. When $Φ$ has a fast computation algorithm, such a dictionary structure allows for more efficient operations with $D$ and its adjoint $D^*$. Due to their genes inherited from the patch-based learning techniques, these methods still suffer from explicit matrix manipulations, restricting their deployment to high dimensional signals.

- **Convolutional dictionary learning (CDL)**, uses convolution as the basic image synthetic operation, $X = \sum_k D_k * Z_k$, where $\{D_k\}_{k=1}^K$ are the set of convolutional kernels learned from training data, and $\{Z_k\}_{k=1}^K$ are the corresponding code maps. The most prominent property of this image model is shift-invariance, contributing two merits over its patch-based counterparts: compact kernel set capable of global image representation [22], and structured dictionary with fast operations using Fast Fourier Transform (FFT) [23]. The convolution operation, however, causes the image boundary problem that can undermine the learning performance. The seminal work [24] by Heide et al. introduced a mask matrix to account for the boundary effect and employed the alternating direction method of multipliers (ADMM) to flexibly and efficiently solve the sparse coding and filter learning sub-problems, in an alternative manner.

As the aforementioned DL algorithms were proposed under the configuration for 2D images, they need some type of adaptation to the 3D data structure of HSI. Some work [4], [8] directly learn the spatial dictionary on each spectral band, neglecting the spectral correlation in HSI. A more widely adopted approach is to reshape the 3D data $\Xi \in \mathbb{R}^{H \times W \times B}$ into a 2D matrix $X \in \mathbb{R}^{R \times N}$, where $N = H \times W$ and $H$, $W$ and $B$ represent the image height, width and channel number, respectively. Then some work [11–13], [5]–[7], [9] conduct DL along the spectral dimension, only capturing the spectral sparsity by expressing a pixel (a column in $X$) as $x_k = Dz_k$. The attempt to partially consider the joint spatial-spectral sparsity is to employ the patch-based DL on small overlapped 2D patches in $X$ [4] or vectorized 3D cubes in $\Xi$ [10], [25], which still cannot guarantee the 3D signature integrity.

Computational complexity and memory consumption entailed by the curse of dimensionality are two critical issues that every DL algorithm has to cope with. As to the practice of patch-based DL in hyperspectral applications, suppose that the dictionary $D$ consists of $N$ atoms, each having a dimension $M$. As discussed previously, the basic operation is matrix-vector multiplication, resulting in a complexity $O(MN)$, which prohibits large atom number as well as large patch size. On the other hand, however, redundant representation requires sufficient quantity of atoms to ensure diversity, which usually needs a tradeoff between the dictionary scale and the computational and memory complexities. In contrast, CDL employs convolution as the basic operation, which can capture local image features through the global sliding process of filters, yielding a small set of convolutional kernels that need to be stored in memory. And the convolution operation also has fast computation by using FFT. Although in methodology concise representation and fast operation are two advantages of CDL over its patch-based counterparts, in practice, how can we design efficient optimization algorithms that can fulfill the CDL training and testing tasks within reasonable memory and running time expenditures? And furthermore, how can we exploit this learned sparse representation in inverse HSI problems and how much benefit can it bring?

To address these problems, this paper introduces the Preconditioned ADMM (PADMM) framework [26], [27] into CDL for HSI and compressive chromo-tomography (CCT) [28], and further proposes two efficient diagonal preconditioner design methods that allow for Hessian free solutions, resulting in substantial algorithm acceleration and memory savings. In particular, we make the following contributions:

- We prove that the subproblems of CDL, sparse coding and filter updating, both can be cast within the PADMM framework, leading to efficient numerical solutions with convergence guarantee. Furthermore, this preconditioning technique is also introduced to the CCT problem, which makes the reconstruction of high dimensional HSI numerically more tractable.

- We present two efficient diagonal preconditioner selection methods, contributing to the proposed Hessian free CDL algorithm, which reduces the computational complexity of involved linear systems to $O(n^2)$, compared with the $O(n^3)$ complexity of Hessian-based matrix inversion approaches, and at the meanwhile removes the storage requirement for the Hessians.

- We experimentally demonstrate the effectiveness of our learned dictionary to CCT, which exhibits robust noise properties, reliable generalization capability, and
outperforming reconstruction performances over analytical transform bases.

The remainder of this paper is organized as follows. A brief survey on CDL is given in Section II. Section III discusses the biconvex optimization framework for hyperspectral CDL. Section IV proposes the algorithm acceleration and memory reduction techniques using PADMM. Application of the learned dictionary to compressive chromo-tomography reconstruction is detailed in Section V. Experimental results are provided in Section VI, and Section VII concludes this paper.

II. RELATED WORK

In this section, we first introduce the classical CDL formulation for single channel images, then extend to the multi-channel scenario.

A. SINGLE CHANNEL CDL

In the literature, a widely used mathematical model for CDL can be formulated as the following optimization problem:

$$\arg\min_{\{D_j\}, \{Z_{j,k}\}} \frac{1}{2} \sum_{j=1}^{J} \left\| \sum_k D_k \ast Z_{j,k} - X_j \right\|_2^2 + \lambda \sum_{j,k} \left\| Z_{j,k} \right\|_1$$

s.t. $\left\| D_k \right\|_F^2 \leq 1, \quad k \in [K].$ (1)

where $D_k \in \mathbb{R}^{h \times w}$ is the $k$th member in a set of $K$ convolutional kernels, $\{X_j\}$ and $\{Z_{j,k}\} \in \mathbb{R}^{H \times W}$ denote the $J$ training images and their corresponding $J \times K$ code maps with respect to $\{D_k\}$, respectively. The symbol $\ast$ is the 2D convolution operator, and $[K]$ represents the set of $1, \ldots, K$. The sparsity of $Z_{j,k}$ is controlled by the regularization weight $\lambda$, and the normalization constraints on $D_k$ help reduce the ambiguity of energy distribution between $Z_{j,k}$ and $D_k$.

Most CDL algorithms employ the alternative minimization strategy to solve problem (1) since it is biconvex w.r.t. the variable set $\{(D_k), \{Z_{j,k}\}\}$. The whole optimization process alternates between the following subproblems, each of which can be solved using ADMM [23], [24], [29], [30] or Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [31], [32].

1) SPARSE CODING SUBPROBLEM (FIXING $\{D_k\}$)

$$\arg\min_{\{Z_{j,k}\}} \sum_{j=1}^{J} \frac{1}{2} \left\| \sum_k D_k \ast Z_{j,k} - X_j \right\|_2^2 + \lambda \sum_{j,k} \left\| Z_{j,k} \right\|_1,$$ (2)

where we follow the notations and definitions on convolution matrix construction in [33] that $D_k \in \mathbb{R}^{(H-h+1)(W-w+1) \times HW}$ is the “cropped” Toeplitz-block-Toeplitz matrix constructed from $D_k$ and $Z_{j,k} \triangleq \text{vec}(Z_{j,k}) \in \mathbb{R}^{HW}$ is the vectorization of $Z_{j,k}$ such that $D_k \ast Z_{j,k} = D_k \ast Z_{j,k}$ keep the "valid" part of their convolution. Likewise, $X_j \triangleq \text{vec}(X_j)$. One should bear in mind that the matrix-vector representation of convolution is just to facilitate algebraic derivations whereas the true computation usually resorts to FFT in the frequency domain.

2) FILTER UPDATE SUBPROBLEM (FIXING $\{Z_{j,k}\}$)

$$\arg\min_{\{D_k\}} \sum_{j=1}^{J} \frac{1}{2} \left\| \sum_k D_k \ast Z_{j,k} - X_j \right\|_2^2 + \lambda \sum_{j,k} \left\| Z_{j,k} \right\|_1,$$ (3)

where $Z_{j,k} \in \mathbb{R}^{(H-h+1)(W-w+1) \times HW}$ is constructed from $Z_{j,k}$ and $D_k \triangleq \text{vec}(D_k)$. Here, the indicator function $\Gamma_C(y)$ takes the place of the filter constraints in (1),

$$\Gamma_C(y) = \begin{cases} 0, & y \in C \\ \infty, & y \in \mathbb{C}, \quad \text{with } C = \{ y \in \mathbb{R}^m : \|y\| \leq 1 \} . \end{cases}$$ (4)

The sparse coding step in (2) can be directly decomposed into $J$ independent subproblems w.r.t. each training image $X_j$. Nevertheless, the filter update step involves all $X_j$, which is much more computationally challenging. A recent work [34] employed the multiconvex optimization strategy [35] to sequentially update each filter and its corresponding sparse codes, in which a block proximal gradient method using a majorizer is used instead of ADMM for lower memory consumption.

Another line of research proposes to use non-convex regularization ($L_0$ norm) and non-convex constraints ($\|D_k\|_F^2 = 1$) [36], [37], which perform the joint optimization of $\{(D_k), \{Z_{j,k}\}\}$, relying on local Lipschitz constant estimation and proper initializations to achieve the global convergence.

B. MULTI-CHANNEL CDL

While plenty of research on single channel CDL have been conducted as discussed above, multi-channel CDL has received relatively few investigations [38], [39]. Basically, when extending CDL from the single channel to the multi-channel scenario, two kinds of approaches can be used: a single channel dictionary combined with independent multi-channel code maps, or a multi-channel dictionary combined with a shared set of code maps. Since the latter case can provide a much more memory efficient representation, we adopt this convolutional representation for HSI in our work.

The mathematical model of multi-channel CDL can be formulated as:

$$\arg\min_{\{D_{k,b}\}, \{Z_{j,k}\}} \sum_{j=1}^{J} \frac{1}{2} \left\| \sum_{k,b} D_{k,b} \ast Z_{j,k} - X_j \right\|_F^2 + \lambda \sum_{j,k} \left\| Z_{j,k} \right\|_1,$$ (5)

where $D_{k,b}$ is the channel $b$ of the filter $k$, $Z_{j,k}$ is the $k^{th}$ code map shared by all $B$ channels $X_{j,b}$, belonging to the training image $X_j$.

When training a convolutional dictionary for HSI, a practical manipulation is to crop HSI in the spatial dimensions into small cubes to alleviate computation and storage problems. Therefore, the image boundary problem should be properly
treated to avoid its undermining influence on dictionary learning. However, existing work [38], [39] on multi-channel CDL employ the model (5) that do not consider the boundary effect. Moreover, their ADMM solutions to the filter updating subproblem employ the Iterated Sherman-Morrison (ISM) formula to solve the linear system, leading to a quadratic complexity on the number of image channels, thereafter not suitable for hyperspectral CDL.

Our work introduces a masking matrix into (5) to account for the boundary effect, and takes advantage of the flexible variable splitting capability of ADMM to solve the sparse coding and filter updating subproblems in an alternative way. In order to acquire further acceleration, we employ PADMM to address the most computationally challenging problem, the linear system solution, by circumventing the explicit Hessian matrix calculation and inversion in the optimization process.

III. BI-CONVEX OPTIMIZATION FRAMEWORK

Now, we are ready to extend the single channel CDL problem to the 2D-representation and 3D-filter mode for HSI to account for the 3D data-cube structure. Such a scheme allows for a memory efficient representation. We also employ the bi-convex optimization framework and take advantage of the algebraic structures of the involved subproblems to effectively reduce the computational complexity.

A. SPARSE CODING SUBPROBLEM

Since convolution boundary handling has been recognized as an important issue for CDL [24], [34], hereafter all images are zero-padded in the spatial dimension to size $(H + h - 1) \times (W + w - 1)$, denoted as $N$. We then introduce a mask matrix $\mathbf{M} \in \mathbb{R}^{N \times N}$ into (2) and rewrite it as follows,

$$\arg \min_{\mathbf{z}_j} \frac{1}{2} \left\| \mathbf{M} \mathbf{D} \mathbf{z}_j - \mathbf{x}_j \right\|^2_2 + \lambda \left\| \mathbf{z}_j \right\|_1, \quad \text{for } j = 1, \ldots, J,$$

(6)

where $\mathbf{z}_j \triangleq \left[ \mathbf{z}_{j,1}^T, \ldots, \mathbf{z}_{j,K}^T \right]^T$ consists of the 2D coefficient maps with respect to the 3D filters in the dictionary $\mathbf{D} \triangleq \left[ \mathbf{D}_1, \ldots, \mathbf{D}_K \right]$ with $\mathbf{D}_k \triangleq \left[ \mathbf{D}_{k,1}^T, \ldots, \mathbf{D}_{k,T}^T \right]^T$.

Problem (6) is convex but nonsmooth, therefore we resort to ADMM for variable splitting. Introducing the auxiliary variables $\mathbf{s}_j = \mathbf{D} \mathbf{z}_j$ and $\mathbf{t}_j = \mathbf{z}_j$, we write the augmented Lagrangian function for (6) as

$$\mathcal{L}_{\mathbf{D}, \mathbf{H}, \mathbf{R}}(\mathbf{s}_j, \mathbf{t}_j, \alpha_s, \alpha_t, \beta_s) \triangleq \frac{1}{2} \left\| \mathbf{M} \mathbf{s}_j - \mathbf{x}_j \right\|^2_2 + \lambda \left\| \mathbf{t}_j \right\|_1 + \left\langle \alpha_s, \mathbf{z}_j - \mathbf{D} \mathbf{z}_j \right\rangle$$

$$+ \left\langle \beta_s, \mathbf{t}_j - \mathbf{z}_j \right\rangle + \rho^{(x)}_{\alpha} \left\| \mathbf{D} \mathbf{z}_j - \mathbf{s}_j \right\|^2_2 + \rho^{(x)}_{\beta} \left\| \mathbf{z}_j - \mathbf{t}_j \right\|^2_2,$$

(7)

where $\alpha_s$ and $\beta_s$ are the dual variables and $\rho^{(x)}_{\alpha}$ and $\rho^{(x)}_{\beta}$ are the respective penalty factors. Then ADMM using the scaled dual variables [40] performs a sequential variable update,

$$\begin{align*}
\mathbf{z}_j^{t+1} &= \left[ \mathbf{D}^T \mathbf{D} + r^{(x)}_{\alpha} \mathbf{I} \right]^{-1} \times \left[ \mathbf{D}^T \mathbf{s}_j + r^{(x)}_{\alpha} \mathbf{t}_j + \alpha_s \mathbf{z}_j \right], \\
\mathbf{s}_j^{t+1} &= \left[ \mathbf{M}^T \mathbf{M} + r^{(x)}_{\alpha} \mathbf{I} \right]^{-1} \times \left[ \mathbf{M}^T \mathbf{x}_j + r^{(x)}_{\alpha} \mathbf{t}_j \right], \\
\mathbf{t}_j^{t+1} &= \mathbf{S}_r \left( \mathbf{z}_j^{t+1} - \beta_s \mathbf{t}_j \right) \triangleq \mathbf{S}_r \left( \mathbf{z}_j^{t+1} - \beta_s \mathbf{z}_j \right), \\
\alpha_s^{t+1} &= \alpha_s + \rho^{(x)}_{\alpha} \left\| \mathbf{D} \mathbf{z}_j - \mathbf{s}_j^{t+1} \right\|^2_2, \\
\beta_s^{t+1} &= \beta_s + \rho^{(x)}_{\beta} \left\| \mathbf{z}_j^{t+1} - \mathbf{t}_j \right\|^2_2,
\end{align*}$$

(8)

where $r^{(x)}_{\alpha} \triangleq \rho^{(x)}_{\alpha} / \rho^{(x)}_{\beta}$, $\xi_j^i \triangleq \xi_j^i + \alpha_s^i$, $\xi_j^i \triangleq \xi_j^i + \beta_s^i$ and $S_r(\cdot) = \text{sign}(\cdot) \cap \max(0, \text{abs}(\cdot) - r)$ is the soft thresholding operator [40].

Since $\mathbf{M}$ is a diagonal matrix, solving for $\mathbf{s}_j$ is cheap. However, the true challenge lies in the matrix inversion $\left[ \mathbf{D}^T \mathbf{D} + r^{(x)}_{\alpha} \mathbf{I} \right]^{-1}$ due to its huge size $KN \times KN$. An existing wisdom on this problem is to resort to the frequency domain solution,

$$\hat{\mathbf{z}}_j^{t+1} = \left[ \hat{\mathbf{D}}^H \hat{\mathbf{D}} + r^{(x)}_{\alpha} \mathbf{I} \right]^{-1} \times \left[ \hat{\mathbf{D}}^H \hat{\mathbf{s}}_j + r^{(x)}_{\alpha} \hat{\mathbf{t}}_j \right],$$

(9)

where $\hat{\mathbf{x}}$ represents the DFT of $\mathbf{x}$. Unfortunately, unlike the single channel CDL whose 2D filters lead to an efficient solution using the matrix inversion lemma [41], the 3D filter structure demands a higher computational cost.

1) VARIABLE REORDERING FOR EQ. (9)

Observing $\hat{\mathbf{D}}^H \hat{\mathbf{D}}$ has the following sparse structure,

$$\hat{\mathbf{D}}^H \hat{\mathbf{D}} = \begin{bmatrix} \sum_b \hat{\mathbf{D}}_{1,b}^H \hat{\mathbf{D}}_{1,b} & \cdots & \sum_b \hat{\mathbf{D}}_{1,b}^H \hat{\mathbf{D}}_{K,b} \\ \vdots & \ddots & \vdots \\ \sum_b \hat{\mathbf{D}}_{K,b}^H \hat{\mathbf{D}}_{1,b} & \cdots & \sum_b \hat{\mathbf{D}}_{K,b}^H \hat{\mathbf{D}}_{K,b} \end{bmatrix},$$

(10)

which is composed of $K \times K$ blocks of diagonal matrices of size $N \times N$. (9) can be decomposed into $N$ independent smaller problems

$$\hat{\mathbf{z}}_{j,n}^{t+1} = \left[ \mathbf{H}_n + r^{(x)}_{\alpha} \mathbf{I} \right]^{-1} \times \left[ \hat{\mathbf{D}}_{j,n}^H \hat{\mathbf{z}}_{j,n} + r^{(x)}_{\alpha} \hat{\mathbf{t}}_{j,n} \right],$$

(11)

where $\hat{\mathbf{D}}_n \in \mathbb{C}^{B \times K}$ samples all the $K$ filters’ frequency spectrum on each of the $B$ channels, at position $n$. $\mathbf{H}_n = \hat{\mathbf{D}}_n \hat{\mathbf{D}}_n^H$ contains all the $n$-th diagonal elements of every block in (10). $\hat{\mathbf{z}}_{j,n}^i \in \mathbb{C}^B$ and $\hat{\mathbf{z}}_{j,n}^i \in \mathbb{C}^K$ are extracted in the same way. Note that the intractable inversion of $\hat{\mathbf{D}}^H \hat{\mathbf{D}} + r^{(x)}_{\alpha} \mathbf{I}$ has been first transformed into the frequency domain then further decomposed into $N$ matrix inversions of much smaller size $K \times K$.

2) COMPLEXITY OF EQ. (11)

The computational overhead of (11) includes three items: i) FFT of $\mathbf{D}$, $O(KBN \log N)$; ii) computation of $\hat{\mathbf{D}}^H \hat{\mathbf{D}}$ in (10), $O(BNK^2)$; and iii) the inversions of $\mathbf{H}_n + r^{(x)}_{\alpha} \mathbf{I}$, $O(NK^3)$. The main complexity of (11) itself lies in the two matrix-vector multiplications, resulting in $O(JNK(K + B))$. Note
that the overhead can be split among the inner iterations of ADMM, with the extra storage of the inversed matrices.

**B. FILTER UPDATE SUBPROBLEM**

The extension of the single channel mode in (3) to the 2D-representation and 3D-filter mode can be formulated as

\[
\arg \min_{\tilde{d}} \frac{1}{2} \| \tilde{M} \tilde{d} - x \|^2 + \Gamma_C (\tilde{d}),
\]

where the whole batch is considered together by defining

\[
\tilde{Z} = \begin{bmatrix}
I_B \otimes Z_{1,1} & \cdots & I_B \otimes Z_{1,K} \\
I_B \otimes Z_{J,1} & \cdots & I_B \otimes Z_{J,K}
\end{bmatrix}, \quad \text{and} \quad \tilde{d} = \begin{bmatrix}
d_1 \\
\vdots \\
d_K
\end{bmatrix},
\]

with \( d_k \triangleq \begin{bmatrix} d_{k,1}^T, \cdots, d_{k,B}^T \end{bmatrix}^T \) and \( \tilde{M} \triangleq \text{blockdiag}[M, \ldots, M] \). Here, the convex set \( C = \{ \tilde{d} \in \mathbb{R}^J : (I - PP^T) \tilde{d} = 0, \| \tilde{d} \| \leq 1 \} \) combines two constraints on the learned filters: their original spatial support and the convex unit-norm ball.

Problem (12) can also be solved using ADMM, resulting in the following iterations

\[
\begin{align*}
d_{k+1}^i &= \left[ \tilde{Z}^T \tilde{Z} + r(\tilde{d}) I \right]^{-1} \times \left[ \tilde{Z}^T \tilde{u}^i + r(\tilde{d}) \tilde{v}^i \right], \\
u^{i+1} &= \left[ \tilde{M}^T \tilde{M} + \rho_a(\tilde{d}) I \right]^{-1} \left[ \tilde{M}^T \tilde{x} + \rho_a(\tilde{d}) \left( \tilde{d}_{k+1}^i + \tilde{\alpha}_d^i \right) \right], \\
\tilde{v}_{k+1}^i &= \text{prox}_{\tilde{G}_C}(\tilde{d}_{k+1}^i + \tilde{\alpha}_d^i), \quad \forall k, \\
\tilde{\alpha}_{d+1}^i &= \tilde{\alpha}_d^i + \tilde{Z} \tilde{u}^{i+1} - \tilde{u}_{d+1}^i, \\
\tilde{\beta}_{d+1}^i &= \tilde{\beta}_d^i + \tilde{d}_{d+1}^i - \tilde{v}_{k+1}^i.
\end{align*}
\]

where \( r(\tilde{d}) \triangleq \rho_b(\tilde{d}) / \rho_a(\tilde{d}) \), \( \sigma_i \triangleq \tilde{u} - \tilde{\alpha}_d^i, \tilde{\vartheta}_i^d \triangleq \tilde{v}_k^i - \tilde{\beta}_d^i \) and the proximal operator \( \text{prox}_{\tilde{G}_C}(\tilde{y}) \) of \( \Gamma_C (\tilde{y}) \) is nothing but the projection onto the convex set \( C \)

\[
\text{prox}_{\tilde{G}_C}(\tilde{y}) = \begin{cases} 
PP^T \tilde{y} / \| PP^T \tilde{y} \|, & \text{if } \| PP^T \tilde{y} \| > 1 \\
PP^T \tilde{y}, & \text{otherwise.}
\end{cases}
\]

Similarly, the most computationally difficult part in (13) is the inversion of matrix \( \tilde{Z}^T \tilde{Z} + r(\tilde{d}) I \), which involves all training samples and thus becomes more challenging than the \( \tilde{Z}^T \tilde{Z} \)-subproblem in (8). The frequency domain solution of the \( \tilde{d} \)-subproblem reads

\[
\tilde{d}_{k+1}^i = \left[ \tilde{Z}^H \tilde{Z} + r(\tilde{d}) I \right]^{-1} \times \left[ \tilde{Z}^H \tilde{\vartheta}_i^d + r(\tilde{d}) \tilde{\vartheta}_i^d \right],
\]

whose efficient computation choice depends on bigger one between the image number \( J \) and the filter number \( K \).

1) **VARIABLE REORDERING FOR EQ. (15)**

\( \tilde{Z}^H \tilde{Z} \) has the following block sparse structure

\[
\tilde{Z}^H \tilde{Z} = \begin{bmatrix}
I_B \otimes \sum_j \tilde{Z}_{j,1}^H \tilde{Z}_{j,1} & \cdots & I_B \otimes \sum_j \tilde{Z}_{j,K}^H \tilde{Z}_{j,K} \\
\vdots & \ddots & \vdots \\
I_B \otimes \sum_j \tilde{Z}_{j,K}^H \tilde{Z}_{j,K} & \cdots & I_B \otimes \sum_j \tilde{Z}_{j,K}^H \tilde{Z}_{j,K}
\end{bmatrix},
\]

which is composed of \( K \times K \) blocks of diagonal matrices of size \( BN \times BN \). Again by variable reordering, we can decompose (15) into BN independent smaller problems

\[
\tilde{d}_{k+1}^{i+1} = \left[ \mathcal{G}_n + r(\tilde{d}) I \right]^{-1} \times \left[ \tilde{Z}_n^H \tilde{\vartheta}_i^{n,b+n} + r(\tilde{d}) \tilde{\vartheta}_i^{n,b+n} \right],
\]

where \( \tilde{Z}_n \in \mathbb{C}^{J \times K} \) samples all the \( K \) filters’ frequency spectrum of each of the \( J \) coefficient maps, at position \( n \).

2) **COMPLEXITY OF EQ. (17)**

The main computational overhead of (17) includes three items: i) FFT of \( Z \), \( O(JKN \log N) \); ii) computation of \( \tilde{Z}^H \tilde{Z} \) in (16), \( O(NK^2) \); and iii) the inversions of \( \mathcal{G}_n + r(\tilde{d}) I \), \( O(NK^3) \). The main complexity of (17) itself lies in the two matrix-vector multiplications, resulting in \( O(NBK(K + J)) \). The overhead can also be split among the ADMM iterations with the extra storage of the inversed matrices.

**C. COMMENTS ON COMPUTATIONAL COMPLEXITY**

We present three comments on the computational complexity of our CDL algorithm:

1) The computational overheads on the matrix reversions in (11) and (17) can be split among the multiple ADMM inner iterations in each subproblem, with the storage cost for the extracted smaller inversed matrices.

2) The decomposed smaller linear system of equations, (11) and (17), are completely independent at all frequency indices. Therefore our algorithm is highly suitable for parallel implementation.

3) The computation of \( \tilde{D}^H \tilde{D} \), \( \tilde{Z}^H \tilde{Z} \), and the corresponding matrix inversions impose quadratic or even cubic dependence on \( K \), rendering the CDL problem still rather computationally expensive.

In the following section, we will introduce the preconditioned ADMM (PADMM) into CDL, which we call PCDL, having a computational complexity of \( O(2NBK) \) (See Section IV.D for complexity analysis).
where \((\rho^*)\) method for HSI, we introduce the preconditioned ADMM and (17) still demand a great amount of computational tationally tractable, the explicit matrix inversions in (11) Although the previously described variable reordering technique in the frequency domain has made CDL compu-
vented by employing the preconditioned ADMM [26].

IV. HESSIAN FREE CDL USING PADMM

Although the previously described variable reordering technique in the frequency domain has made CDL computationally tractable, the explicit matrix inversions in (11) and (17) still demand a great amount of computational efforts as well as the corresponding memory storage. In order to acquire a lightweight and efficient CDL optimization method for HSI, we introduce the preconditioned ADMM (PADMM) [26], [27] into CDL, which we call PCDL. The proposed PCLD does not need explicit calculation or storage of the Hessians \(D^T D\) and \(Z^T Z\), eliminating the quadratic or higher order computational complexity on the problem scale \((N, B, K, J)\), as well as the quadratic memory requirement on filter numbers \(K\). The framework of PCDL is shown in Fig.1.

A. PRECONDITIONED ADMM

Classical ADMM algorithms solve the following problem:

\[
\min_{p, q \in \mathcal{Q}} F(p) + G(q), \quad s.t. \quad Ap + Bq - c = 0, \tag{18}
\]

where \(\mathcal{P}, \mathcal{Q}\), and \(\mathcal{W}\) are Hilbert spaces, \(F(\cdot) : \mathcal{Q} \to \mathbb{R}\) and \(G(\cdot) : \mathcal{Q} \to \mathbb{R}\) are proper, convex and lower semi-continuous functions, \(c \in \mathcal{W}\) and \(A : \mathcal{P} \to \mathcal{W}, B : \mathcal{Q} \to \mathcal{W}\) are linear and continuous mappings.

The augmented Lagrangian of problem (18) reads:

\[
\mathcal{L}_\rho (p, q, \gamma) = F(p) + G(q) + \langle \gamma, Ap + Bq - c \rangle + \frac{\rho}{2} \|Ap + Bq - c\|^2, \tag{19}
\]

where \(\gamma \in \mathcal{W}\) is the dual variable, and \(\rho\) is the penalty parameter. \(\mathcal{P}^*\) and \(\mathcal{Q}^*\) denote the adjoint and the subgradient operators, respectively. The optimality condition of (19) involves the evaluations of \((\rho A^* A + \partial F)^{-1}\) and \((\rho B^* B + \partial G)^{-1}\), which are computationally challenging and can be circumvented by employing the preconditioned ADMM [26].

Lemma 1. (PADMM, [26, Theorem 3.1]) Assume problem (19) has a saddle point, then the following sequence \((p^i, q^i, \gamma^i)\) converges to an optimal solution of problem (19):

\[
p^{i+1} \leftarrow \arg \min_p \mathcal{L}_\rho (p, q^i, \gamma^i) + \frac{\rho}{2} \|p - p^i\|^2_{Q - \rho A^* A}, \tag{20}
\]

\[
q^{i+1} \leftarrow \arg \min_q \mathcal{L}_\rho (p^{i+1}, q, \gamma^i) + \frac{1}{2} \|q - q^i\|^2_{L - \rho B^* B}, \tag{20}
\]

\[
\gamma^{i+1} \leftarrow \gamma^i + \rho \left( Ap^{i+1} + Bq^{i+1} - c \right), \tag{20}
\]

if the following conditions hold:

(a) \(\rho A^* A + \partial F\) and \(\rho B^* B + \partial G\) are strongly monotone.

(b) \(Q - \rho A^* A, L - \rho B^* B > 0\), where \(Q \in \mathcal{P}\) and \(L \in \mathcal{P}\) are linear, bounded, self-adjoint, and positive definite operators.

(c) \(A\) or \(B\) is compact.

When \(Q\) and \(L\) are chosen properly, (20) can be solved efficiently due to the added proximal regularization items with the form \(\frac{1}{2} \|x - x^*\|^2_{\mathbb{L}} = L(x - x^*), x - x^*\). This preconditioning strategy has also been proposed in the primal-dual framework [42], [43]. Below, we will demonstrate how to employ this method to reduce the numerical burden on the CDL algorithm.

B. PRECONDITIONED SPARSE CODING

Now we present the first contribution of this paper.

Proposition V.2. If we take a bounded and positive definite matrix \(Q > \rho A^* A\) and \(L = I\), then the CSC subproblem (6) can be cast within the PADMM framework, resulting in the preconditioned solution:

\[
z_j^{i+1} = \arg \min_{z_j} \mathcal{L}_\rho (z_j, s_j, t_j, \alpha_j, \beta_j) + \frac{1}{2} \|z_j - z_j^i\|^2_{Q - \rho A^* A D^T D} + \frac{\rho}{2} \|G(z_j) - Dz_j^i\|^2_{\rho B^* B}, \tag{21}
\]

Proof: We first transform the augmented Lagrangian in (7) so as to comply with the standard form in (19), by setting:

\[
\begin{align*}
p_A & \triangleq [z_j], q & \triangleq [s_j, t_j], \\ \gamma & \triangleq [\alpha_j, \beta_j], A & \triangleq [\sqrt{\rho A^* A}, \sqrt{\rho B^* B}], B & = -I, c = 0, F(p) & \equiv 0, G(q) \equiv 0, \\ \mathcal{P} & \triangleq \left\{ z_j \in \mathbb{R}^D : \|M D s_j - x_j\|^2 + \lambda \|t_j\|_1 \right\}, \text{ and } \rho = 1. \end{align*}
\]

Under this configuration, we verify the three conditions in Lemma V.1 for CSC.

(a) Since \(\rho = 1\) and \(F(p) \equiv 0\) in our setting, we have \(\rho A^* A + \partial F = A^* A = \rho A^* A D^T D + \rho B^* B > 0\), yielding its strong monotonicity [44] (Definition 22.1).

Similarly, we have \(\rho B^* B + \partial G = I + \partial G\), where the subgradient \(\partial G\) is monotone because \(G\) is proper and convex [44] (Example 20.3). Thus the identity operator \(I\) ensures the strong monotonicity of \(\rho B^* B + \partial G\).

(b) Since we take \(Q > \rho A^* A D^T D\) and \(L = I\), directly we have \(Q - \rho A^* A = Q - \rho A^* A D^T D + \rho B^* B > 0\). And \(L - \rho B^* B = 0\) actually means we use ADMM updates for the auxiliary variable \(q\).

(c) Our construction of \(A\) and \(B\) leads to their compactness.

Finally, the right hand side of Eq. (21) is the closed-form solution of the least squares problem thereof.
By the Parseval’s Theorem, we can also precondition the \( \hat{z} \)-subproblem in the frequency domain,
\[
\hat{z}_{j+1} = \arg \min_{\hat{z}_j} \mathcal{L}_{\rho_a \rho_b} (\hat{z}_j; \hat{v}_j, \hat{d}_j, \hat{z}_j)
+ \frac{1}{2} \left\| \hat{z}_j - \hat{z}_{j-1} \right\|^2_{Q^{-\rho_b} \hat{D}^H \hat{D}}
= \left[ Q + \rho_b^2 I \right]^{-1} \left[ \rho_a \hat{D}_n^H (\hat{z}_j - \hat{D}_n \hat{z}_{j-1}) \right]
+ \rho_b^2 \hat{z}_j + Q \hat{z}_{j-1}.
\] (22)

Using the same variable reordering manipulation as in (11), the reordered version of (22) at each frequency index \( n \) reads
\[
\hat{z}_{j+1, n} = \left[ Q_n + \rho_b^2 I \right]^{-1} \left[ \rho_a \hat{D}_n^H (\hat{z}_j - \hat{D}_n \hat{z}_{j-1}) \right]
+ \rho_b^2 \hat{z}_j + Q \hat{z}_{j-1, n}.
\] (23)

If we choose \( Q_n \) to be diagonal matrices, then the main complexity of (23) lies in the two matrix-vector multiplications of size \( J \times K \), resulting in \( \mathcal{O} (2JNK) \). Next we present two approaches to selecting feasible diagonal matrices \( Q_n \), which does not require the computation of the Hessian \( \hat{D}^H \hat{D} \).

C. DIAGONAL PRECONDITIONER SELECTION

Next, we present the second contribution of this paper.

1) RICHARDSON PRECONDITIONER

For \( n = 1, \ldots, N \), let \( \theta_n^{(2)} = \bar{\sigma}^2 (\hat{D}_n) \) be the spectral radius of \( \hat{D}_n^H \hat{D}_n \), and \( \epsilon \) a small constant. When taking
\[
Q_n = \rho_a \left( \theta_n + \epsilon \right) I,
\] (24)
we have \( Q_n - \rho_a \hat{D}_n^H \hat{D}_n > 0 \), which corresponds to the simplest Richardson preconditioner.

To avoid the computation of \( \hat{D}_n^H \hat{D}_n \), we use singular value decomposition (SVD) to get the biggest singular value of \( \hat{D}_n \), denoted as \( \sigma_1 (\hat{D}_n) \). Since \( \hat{D}_n \) is of size \( B \times K \), and the SVD of an \( B \times K \) matrix has complexity \( \mathcal{O} (BK \min(K, B)) \) when only singular values are desired [45], [46], computing all \( r_n \) has the complexity \( \mathcal{O} (NBK \min(K, B)) \). A byproduct of the block-wise SVD is the spectral radius of \( \hat{D}_n^H \hat{D}_n \), \( \theta_n^{(2)} = \max_n \left\{ \bar{\sigma}^2 (\hat{D}_n) \right\} \), which is true because neither the FFT nor the permutation matrix changes the eigenvalues of \( \hat{D}_n^H \hat{D}_n \).

2) DIAGONAL MAJORIZING PRECONDITIONER

The following proposition provides an efficient way to constructing a diagonal majorizing preconditioner.

Proposition V.3. Let \( \hat{D} \in \mathbb{C}^B \times K \), and construct the diagonal matrix \( Q = \text{diag} (q_1, \ldots, q_K) \), where
\[
q_k = \sum_{b=1}^{B} |D_{b,k}| c_b, \quad \text{and} \quad c_b = \sum_{k'=1}^{K} |D_{b,k'}|.
\] (25)

Then \( Q \succ D^H D \).

Proof: One can find that \( \{q_k\} \) are nothing but the row sums of \( |D^H| \). Therefore the Hermitian matrix \( Q - D^H D \) is diagonally dominant with nonnegative diagonal entries. Then by the Gershgorin’s circle theorem, every eigenvalue \( \lambda (Q - D^H D) \geq 0 \), which completes the proof. \( \square \)

Note that constructing \( Q \) using (25) has the complexity \( \mathcal{O}(BK) \), and thus is more efficient than using SVD.

D. PRECONDITIONED FILTER UPDATE

As the ADMM solution for the filter update subproblem has similar structure as in the scenario of CSC, the arguments in Sec. IV.C also apply to the filter update subproblem. Here we directly give the preconditioned solution for the filter update
\[
\hat{d}_{h,n}^{t+1} = \left[ Q_n + \rho^d \right]^{-1} \left[ \rho_a \hat{Z}_n^H (\hat{d}_{h,n}^{t+1} - \hat{D}_n \hat{d}_{h,n}^t) \right]
+ \rho^d \hat{d}_{h,n}^t + Q_n \hat{d}_{h,n}^t.
\] (26)

The main computational complexity of (26) lies in the two matrix-vector multiplications of size \( B \times K \), resulting in complexity \( \mathcal{O} (2JNBK) \) for the total \( JN \) blocks.

E. SUMMARY OF PCDL

Our technical avenue for acceleration can be divided into four stages:

- Firstly, the Fourier Trick is exploited to decompose the original large linear system involving \( \hat{D}^T \hat{D} (KN \times KN) \) and \( \hat{Z}^T \hat{Z} (KB \times KN) \) into respectively \( N \) and \( NB \) independent smaller linear systems of scale \( K \times K \), which makes direct matrix inversion feasible.

- Secondly, we further employ PADMM to precondition each frequency block rather than using a global preconditioner, which results in tighter bounds for the preconditioners and thus faster convergence rate.

- Thirdly, we design diagonal preconditioning matrices whose complexity are only \( \mathcal{O} (NBK) \) for sparse coding and \( \mathcal{O} (NKJ) \) for filter update, respectively. Comparatively, the direct inversions are \( \mathcal{O} (NK^3) \).

- Fourthly, by using diagonal preconditioners, neither computation nor storage of the Hessians \( \hat{D}^T \hat{D} \) and \( \hat{Z}^T \hat{Z} \) is required, which further save \( \mathcal{O} (BNK^2) \) and \( \mathcal{O} (NK^2) \), respectively.

The whole procedure of PCDL is shown in Algorithm 1.

V. COMPRESSIVE CHROMOTOGRAPHIC IMAGING RECONSTRUCTION USING CSC

As the third main contribution of this paper, we demonstrate the usefulness of our learned convolutional dictionary to CCT reconstruction.

A. IMAGING MODEL

Chromato-tomographic hyperspectral imaging spectrometer (CTHIS) was first proposed by Mooney, etc. in [47] to achieve a high optical throughput, which is of significant interest for weak target hyperspectral measurement and also for hyperspectral imager development with small form factor.
A schematic of CTHIS is shown in Fig. 2, where a rotating direct view prism (DVP) is implanted into the collimated space to effectively displace the spectral images on the focal plane array (FPA) with the wavelength dependent shifts in \( w \) and \( h \) directions,

\[
\begin{aligned}
\delta_w (\omega, \varphi) &= k (\omega - \omega_c) \cos \varphi, \\
\delta_h (\omega, \varphi) &= k (\omega - \omega_c) \sin \varphi,
\end{aligned}
\]  

where \( k \) is the dispersion coefficient of the DVP, \( \omega \) is the wavelength, \( \omega_c \) and \( \varphi \) denote the undeviated wavelength and the rotation angle of the DVP, respectively. By calibration, the optical parameters \( k \) and \( \omega_c \) are transformed into pixel number and spectral channel number, respectively.

The spectral images shifted according to (27) will be integrated along the chromatic dimension by the FPA [47],[28], which mathematically follows the Radon transform and is comparable to the parallel beam X-ray tomography, hence the name of chromo-tomography. The original reconstruction algorithm of CTHIS amounts to solving an overdetermined linear system of equations in the Fourier domain, while this work exploits the compressed sensing theory [48],[49] to pursue the compressive chromo-tomography reconstruction from several chromo-tomographic shots fewer than the spectral band number.

Denote the lexicographically stacked representation of the original datacube by \( x \in \mathbb{R}^{H \times W \times B} \), and the measurement vector captured at DVP angle \( \varphi_i \) by \( g_i \in \mathbb{R}^{H \times W} \). Then the discrete imaging model of CCT can be formulated as

\[
g_i = A_i x, \tag{28}\]

where the forward system matrix \( A_i \triangleq [A_{i1}, \ldots, A_{iB}] \) stands for the summation of all the \( B \) shifted spectral channels, and \( A_{ib} \) is a \( b \in [B] \) implements the spatial shift of channel \( b \) [28],

\[
A_{ib} = F^H \text{diag} \left( \left[ e^{-j2\pi (w \delta_{w}^{(b,l)} + h \delta_{h}^{(b,l)}) / H} \right] \right)_{(h,w)=((1,1))}, \tag{29}\]

where \( F \) is the Fourier transform matrix, \( \delta_{w}^{(b,l)} \) and \( \delta_{h}^{(b,l)} \) are the image shifts caused by dispersion in (27).

Finally, suppose we have \( L \) measurements defined in (28), then they can be concatenated to form

\[
g = Ax, \tag{30}\]

where \( g \triangleq [g_1^T, \ldots, g_L^T]^T \) and \( A \triangleq [A_1^T, \ldots, A_L^T]^T \).

### B. COMPRESSIVE RECONSTRUCTION ALGORITHM

It’s well known that CSC is a rather effective representation method for the high-frequency (HF) components in images, which has been demonstrated in different image processing tasks [50]–[56]. When dealing with the low-frequency (LF) components in single channel CSC applications, a common method is to add a low-pass Gaussian filter [24],[29],[57] or a Dirichlet function [51],[53] into the learned filter set. However, a direct extension of this manipulation to hyperspectral images cannot effectively cope with the variations among spectral channels.

In this section, we first acquire an HSI estimate via a least square-total variation (LS-TV) algorithm, then retrieve its LF component which is further fed into a LS-TV-CSC framework for the HF component enhancement.

1) **LS-TV RECONSTRUCTION**

We first solve the following LS-TV problem for an intermediate image reconstruction result,

\[
\arg\min_{x} \frac{1}{2} \left\| Ax - g \right\|^2 + \lambda \| x \|_{TV} . \tag{31}\]
where $A$ and $g$ are defined in (30), and $\|x\|_{TV} \|\Omega x\|_2$, with $\Omega \triangleq I_B \otimes \tilde{\Omega}$. Thereof, $\otimes$ is the Kronecker product and $\tilde{\Omega} \in \mathbb{R}^{2HW \times HW}$ is the total finite difference operator [58].

Introducing the auxiliary variable $s = \Omega x$, the ADMM solution to (31) is obtained through the following alternations:

$$
x^{i+1} = \arg \min_x \frac{1}{2} \left\| Ax - g \right\|_2^2 + \frac{\rho_A}{2} \left\| \Omega x - s + \alpha' \right\|_2^2,
$$

$$
s^{i+1} = \arg \min_s \lambda \left\| s - \alpha \right\|_2^2,
$$

$$
\alpha^{i+1} = \alpha^i + \rho_A \left( \Omega (s^{i+1} - s^i) \right).
$$

where $\rho_A$ is the penalty factor and $\alpha$ is the dual variable.

**x-subproblem:** The $x$-subproblem in (32) involves the big inverse matrix $[A^T A + \Omega^T \Omega]^{-1}$, so we employ the preconditioning trick introduced in Section IV and get the preconditioned $x$-subproblem as follows:

$$
\arg \min_x \frac{1}{2} \left\| Ax - g \right\|_2^2 + \frac{\rho_A}{2} \left\| \Omega x - s + \alpha' \right\|_2^2
$$

$$
+ \frac{1}{2} \left\| x - x^i \right\|_{Q_1}^2 + \frac{1}{2} \left\| x - x^i \right\|_{Q_2}^2,
$$

where $Q_A = N_A - A^T A > 0$ and $Q_\Omega = N_\Omega - \rho_A \Omega^T \Omega > 0$ are required.

Note that by the definition of $A'_i$ in (29), $\hat{A}^H \hat{A}$ has the same structure as $\hat{D}^H \hat{D}$ in (10). Thus a diagonal preconditioner $N_A$ can be obtained in the same way as the filter update problem does using (24) or (25). As to $\Omega^T \Omega \triangleq I_B \otimes \tilde{\Omega}^T \tilde{\Omega}$, we use an established bound on the operator norm $\|\tilde{\Omega}\|_2 \leq \sqrt{8}$ [59], leading to a diagonal $N_\Omega = 8 \rho_A I$. Therefore, the solution of (33) reads,

$$
x^{i+1} = [N_A + N_\Omega]^{-1} \left[ A^T (g - Ax^i) + \rho_A \Omega^T (s^i - \alpha'/\rho_A - \Omega x^i) + (N_A + N_\Omega) x^i \right].
$$

**s-subproblem:** The solution of $s$ in (32) is given by the 2D shrinkages per pixel $n \in [HW]$ and per channel $b \in [B]$,

$$
s_{b, n}^{i+1} = \max \left\{ \|s\|_2 - \lambda/\rho_A, 0 \right\} \|\theta\|_2,
$$

where $\theta = \tilde{\Omega} s_{b, n}^{i+1} + \alpha'/\rho_A$, and $\tilde{\Omega} \in \mathbb{R}^{2 \times HW}$ extracts the horizontal and vertical differences at pixel $n$ in the spectral band image $s_b$.

Once the stopping criterion of (32) is reached, we take the estimate $\hat{x}$ as the intermediate result and extract its LF part $v$ by a low-pass filtering,

$$
v = G(\hat{x}),
$$

where $G(\cdot)$ stands for a low-pass Gaussian filtering.

2) LS-TV-CSC ENHANCEMENT

The HF parts of compressive measurements are extracted by

$$
g_b = g - Av.
$$

Since inaccurate filters and structure loss may exist in the learned dictionary $D$ [50], we augment the CSC prior with a TV regularization to surpass the possible artifacts, and reconstruct the HF part by solving the following problem,

$$
\arg \min_z \frac{1}{2} \| A D z - g_b \|_2^2 + \mu \| D z \|_{TV} + \tau \| z \|_1
$$

where $z$ is the CSC map corresponding to the HF components of the unknown hyperspectral image $x$.

Introducing auxiliary variables $u = D z$ and $t = z$, the ADMM solution to (38) is obtained through the following alternations:

$$
x^{i+1} = \arg \min_x \frac{\rho_A}{2} \| x - t^i + y^i \|_2^2
$$

$$
+ \frac{\rho_B}{2} \| D z^i - u + \beta^i \|_2^2,
$$

$$
u^{i+1} = \arg \min_u \frac{\rho_B}{2} \| A u - g_b \|_2^2 + \mu \| u \|_{TV}
$$

$$
+ \frac{\rho_B}{2} \| D z^i - u + \beta^i \|_2^2,
$$

$$
t^{i+1} = \arg \min_t \| t \|_1 + \left( y^i, z^{i+1} - t \right)
$$

$$
+ \frac{\rho_B}{2} \| z^{i+1} - t \|_2^2,
$$

$$
\beta^{i+1} = \beta^i + \rho_B \left( D z^{i+1} - u^{i+1} \right),
$$

$$
y^{i+1} = y^i + \rho_A \left( z^{i+1} - t^{i+1} \right).
$$

**z-subproblem:** One can find the $z$-update in (39) is nothing but the single image version of the $z$-solution in (8) for CSC. Therefore, here we directly give its preconditioned solution similar to (21),

$$
z^{i+1} = \left[ N_D + \rho_B I \right]^{-1} \left[ \rho_B D^T \left( u^i - \beta^i/\rho_B - D z^i \right) + \rho_B \left( t^i - y^i/\rho_B \right) \right].
$$

where $N_D > \rho_B D^T D$ can be constructed using (24) or (25) in the same way, and the frequency solution of (23) is also applicable to (40).

**t-subproblem:** The $Z$-update is given by the soft thresholding,

$$
t^{i+1} = \text{sign}(b) \max \left( 0, \text{abs}(b) - \tau/\rho_B \right),
$$

where $b = z^{i+1} + y^i/\rho_B$.

**u-subproblem:** The $u$-update needs to solve a proximal regularized LS-TV problem similar to (31), and we leave its solution to the next subsection.

Once the stopping criterion of (39) is reached, we use the optimized CSC map $\hat{z}$ to synthesize the HF components and
Algorithm 2 Compressive CCT Reconstruction

1: **Input:** chromo-tomography measurements \( \mathbf{g} \), Gaussian filter \( \mathbf{G} \), 3D CSC filters \( \mathbf{D} \), CCT system matrix \( \mathbf{A} \) and TV operator \( \mathbf{\Omega} \), positive parameters \( \lambda, \mu, \tau, \rho_\alpha, \rho_\beta, \rho_\gamma, \rho_\phi \).

2: **Initialize:** auxiliary variable \( s, e, u, t \); dual variable \( \alpha^0, \phi^0, \beta^0, \gamma^0 \).

3: **LS-TV Reconstruction:**

4: **Repeat:**
5: Update \( \mathbf{x}^{i+1} \) via EQ. (34).
6: Update \( \mathbf{s}^{i+1} \) via EQ. (35).
7: Update the variable \( \alpha^{i+1} \) in EQ. (32).
8: **End** when reaching the maximum iteration number \( I_L \).
9: Get the LF part \( \mathbf{v} \) via EQ. (36).

10: **LS-TV-CSC Enhancement:**
11: Get the HF measurements \( \mathbf{g}_h \) via EQ. (37).
12: **Repeat:**
13: Update \( \mathbf{z}^{i+1} \) via EQ. (40).
14: Update \( \mathbf{t}^{i+1} \) via Algorithm 2.
15: Update variables \( \beta^{i+1} \) and \( \gamma^{i+1} \) in EQ. (39).
16: **End** when reaching the maximum iteration number \( I_L \).
17: **Get the final reconstruction via EQ.** (42).

combine it with the LF part \( \mathbf{v} \) obtained in (36), getting the final reconstructed hyperspectral image

\[
\hat{\mathbf{x}} = \mathbf{D} \mathbf{z} + \mathbf{v}. \tag{42}
\]

The overall algorithm for compressive chromo-tomographic reconstruction is summarized in Algorithm 2.

3) SOLUTION OF U-SUBPROBLEM

Introducing the auxiliary variable \( e = \mathbf{\Omega} u \), we conduct an inner ADMM alteration for the \( u \)-subproblem in (39):

\[
\begin{align*}
\mathbf{u}^{i+1} & = \arg \min \frac{1}{2} \| \mathbf{A} \mathbf{u} - \mathbf{g}_h \|_2^2 + \frac{\rho_\beta}{2} \| \mathbf{D} \mathbf{z} - \mathbf{u} + \frac{\beta^i}{\rho_\beta} \|_2^2 \\
& \quad + \frac{\rho_\phi}{2} \| \mathbf{\Omega} \mathbf{u} - e^i + \frac{\phi^i}{\rho_\phi} \|_2^2 , \\
\mathbf{e}^{i+1} & = \arg \min _\mathbf{e} \| \mathbf{e} \|_2 + \frac{\rho_\gamma}{2} \| \mathbf{\Omega} \mathbf{u}^{i+1} - \mathbf{e} + \frac{\gamma^{i+1}}{\rho_\gamma} \|_2^2 , \\
\phi^{i+1} & = \phi^i + \phi \left( \mathbf{\Omega} \mathbf{u}^{i+1} - e^{i+1} \right).
\end{align*}
\tag{43}
\]

where \( \rho_\phi \) is the penalty factor and \( \phi \) is the dual variable.

**u-update:** The \( u \)-update in (43) has similar structure to the \( x \)-subproblem in (32). Here, we also add two preconditioning items \( \frac{\lambda}{2} \| \mathbf{u} - \mathbf{u}^c \|_{\mathbf{G}_\lambda}^2 \) and \( \frac{\tau}{2} \| \mathbf{u} - \mathbf{u} \|_{\mathbf{Q}_\tau}^2 \) to the \( u \)-update objective function so that \( A^T \mathbf{A} \) and \( \Omega^T \Omega \) can be canceled out for efficient computation. The resultant solution is given by

\[
\mathbf{u}^{i+1} = \left[ \mathbf{N}_\mathbf{A} + \rho_\beta \mathbf{I} + \mathbf{N}_\mathbf{\Omega} \right]^{-1} A^T \left( \mathbf{g}_h - \mathbf{A} \mathbf{u}^i \right)
\]

Algorithm 3 Solution of the u-Subproblem in EQ. (39)

1: **Input:** high frequency measurements \( \mathbf{g}_h \), matrices \( \mathbf{A} \) and \( \mathbf{\Omega} \), parameters \( \rho_\beta, \rho_\phi \), variables \( \beta^i \) and \( \gamma^i \).

2: **Initialize:** auxiliary variable \( \mathbf{e} \), dual variable \( \mathbf{\phi} \), parameter \( \mu \), iteration index \( j = 1 \).

3: **Repeat:** \{Iterations: \( J \} \; (J \geq 1)
4: Update \( \mathbf{u}^{i+1} \) via EQ. (44).
5: Update \( \mathbf{e}^{i+1} \) via EQ. (45).
6: Update the variable \( \phi^{i+1} \) in EQ. (43).
7: **End** when reaching the maximum iteration number \( J \).
8: **Output:** \( \mathbf{u}^{i+1} = \mathbf{u}^J \).

\[
+ \rho_\beta \left( Dz^i - \frac{\beta^i}{\rho_\beta} \right) + \rho_\phi \Omega^T \left( e^i - \frac{\phi^i}{\rho_\phi} - \Omega u \right) + (N_A + N_\Omega) \mathbf{u}^J.
\tag{44}
\]

where \( N_A \) and \( N_\Omega \) are same with those in (34).

**e-update:** The solution of \( \mathbf{e} \) in (43) is parallel to the solution in (35),

\[
\mathbf{e}^{j+1}_{\nu,n} = \max \left\{ \| \mathbf{e} \|_2 - \mu / \rho_\phi, 0 \right\} \mathbf{\theta} / \| \mathbf{\theta} \|_2, \tag{45}
\]

where \( \mathbf{\theta} = \tilde{\Omega}_n \mathbf{u}^{j+1}_n + \alpha^i / \rho_\alpha \).

The ADMM alteration for the \( u \)-subproblem is detailed in Algorithm 3.

VI. EXPERIMENTS AND ANALYSIS

In this section, we first verify the convergence performance of the proposed PCDL algorithm with different dictionary parameter settings. Then numerical simulation experiments using the learned convolutional dictionaries are carried out to demonstrate the effectiveness of CSC to compressive chromo-tomography reconstruction.

A. CDL EXPERIMENTAL SETUP

All CDL algorithms in this section are with Matlab implementations and ran on a workstation with an Intel Xeon CPU E5-2620 and 64GB RAM. All comparison algorithms were fed with identical randomly initialized feature maps and filters for fair comparison.

1) ALGORITHM NAMING CONVENTIONS

We compare four ADMM-based implementations of CDL for HSI with convex constraints, including:

- **HCDL:** This is the implementation of the Hessian-based algorithm described in Section III, which requires the calculation and inversion of Hessian matrices and can be regarded as a 3D extension of the fast and flexible CDL in [24]. Since accurate matrix inversion is directly employed, this version can serve as a benchmark for convergence rate.

- **HCDL-greedy:** This is a greedy version of HCDL that only performs one ADMM iteration in each step, which can be regarded as a 3D extension of the fast CDL [23] and the efficient CSC using ISM [32].
• **PCDL1**: This is the implementation of the proposed Algorithm 1, using the Richardson type preconditioner in Eq. (24).

• **PCDL2**: This is the implementation of the proposed Algorithm 1, using the diagonal majorizing preconditioner in Eq. (25).

2) **HSI DATASET SPECIFICATION**

We use the Columbia CAVE dataset in [60] as our training set. The original HSI is of size 512 × 512 × 31, covering a spectral range from 400nm to 700nm. We downsampled the original images by a factor of 2 and cropped them into small samples of size 128 × 128 × 31. All samples were normalized to [0, 1] and then low-pass filtered by a 13 × 13 Gaussian filter with the standard deviation parameter \( \sigma \) empirically set to 4.773. Finally, their LF components were removed and the resultant HF parts were used for CDL.

3) **CDL ALGORITHM PARAMETER SETTING**

The parameters in Algorithm 1 were set as follows:

\[
\lambda = 1, \quad \rho_\alpha = \rho_\xi = 60/5000T, \quad \text{and} \quad \rho_\beta = \rho_\gamma = 60/500T,
\]

where \( T \) is the maximum intensity of the input images [61]. We used \( S = V = 5 \) inner iterations for each filter and feature map update, respectively.

4) **COMPRESSION RECONSTRUCTION ALGORITHM PARAMETER SETTING**

The parameters in Algorithm 2 were set as: \( \lambda = \mu = 0.7, \tau = 5, I_L = 200, I_H = 200, \text{and} \rho_\alpha = \rho_\beta = \rho_\gamma = \rho_\phi = 20/50T. \) And the maximum iteration number was \( J = 3 \) in Algorithm 3. The parameters in Algorithm 2 and 3 were empirically selected and kept constant throughout the experiments in this section.

5) **STOPPING CRITERION**

Iterations are terminated if either of the following condition is satisfied:

\[
F(d^{(l+1)}, z^{(l+1)}) \geq F(d^{(l)}, z^{(l)}) \quad \text{and} \quad F(d^{(l+1)}, z^{(l+1)}) \\
\geq F(d^{(l)}, z^{(l)})
\]

or

\[
\|d^{(l+1)} - d^{(l)}\|_2 \leq \|d^{(l+1)}\|_2 < \text{tol} \quad \text{and} \quad \|z^{(l+1)} - z^{(l)}\|_2 < \text{tol},
\]

where \( F(d, z) \) is the objective function defined in (5), and \( l \) is the iteration index in the outer loop in Algorithm 1. The parameter \( \text{tol} \) was set to \( 1e^{-3} \).

**B. CDL CONVERGENCE COMPARISON**

In this experiment, we conduct two CDL tasks with different scales for every algorithm described above. The first one uses a training set of moderate size with \( J = 32 \) samples to train \( K = 25 \) filters. And the second one learns \( K = 50 \) filters from \( J = 100 \) samples. The cost function value vs. iteration number curves are drawn in Fig.3 (a) and (c), which shows that the two PCDL algorithms converge with slightly slower rates compared with the HCDL versions that use exact matrix inversions. However, the corresponding running time curves drawn in Fig.3 (b) and (d) reveal that PCDL can achieve same optimization performance with HCDL but with much faster speed. This experiment verifies the reliable convergence property of PCDL compared with those resorting to exact Hessian inversions. Both the runtime and the computational complexity analysis are presented in Table 1 for a comprehensive comparison.

**C. LEARNED FILTERS**

The 3D filters learned by PCDL2 in the two tasks are displayed in Fig.4. We assemble the filters in each channel for observing their intra-channel diversity, while also provide a 2D colorful version by dimensionality reduction to reveal the chromatic variation among the 3D filters.

This experiments shows that the proposed CDL method successfully captures the spatial-spectral features hidden in the HSI samples. Note that in the colorful display of the 25-filter dictionary, there exist some grey ones, which correspond to common features among all spectral channels and thus have constant appearance. However, when it comes to the 50-filter dictionary, we cannot observe such phenomenon maybe because the designated bigger filter number allows for the dispersion of similar spectral variations into more filters. Therefore, an interesting question arises: Is it true that more filters lead to better results for inverse problems? We
investigate this problem through the following experiments on CCT reconstruction.

D. CCT RECONSTRUCTION PERFORMANCE

1) EFFECT OF FILTER NUMBERS

This experiment was designed to compare the CCT reconstruction performances using the aforementioned two dictionaries. In order to take into account the generalization capability of the dictionaries, two test sets were compared: Set1, consisting of 4 images ((1)-(4) in Fig.5) randomly picked up from the training set; Set2, consisting of 4 images ((10)-(13) in Fig.5) randomly picked up from the ICVL dataset [62]. All test hyperspectral images are of size $256 \times 265 \times 31$. Using each of the test images, we simulated multiple 2D chromo-tomographic measurements via the imaging model (30) as the inputs of Algorithm 2.

| Method     | Filter Update                                                               | Sparse Coding                                         | Runtime (s) (25 filters, 32 samples) | Runtime (s) (50 filters, 100 samples) |
|------------|----------------------------------------------------------------------------|-------------------------------------------------------|------------------------------------|--------------------------------------|
| HCDL       | $O\left( \frac{1}{4}K \log N + \frac{1}{4}K \log J \right)$ + $O\left( \frac{1}{4}K \log N \right)$ + $O\left( \frac{1}{4}K \log J \right)$ & $O\left( \frac{1}{4}K \log N \right)$ + $O\left( \frac{1}{4}K \log J \right)$ + $O\left( \frac{1}{4}K \log N \right)$ + $O\left( \frac{1}{4}K \log J \right)$ & 7174s | 31465s |
| HCDL-greedy| $O\left( \frac{1}{4}K \log N \right)$ + $O\left( \frac{1}{4}K \log J \right)$ + $O\left( \frac{1}{4}K \log N \right)$ + $O\left( \frac{1}{4}K \log J \right)$ & $O\left( \frac{1}{4}K \log N \right)$ + $O\left( \frac{1}{4}K \log J \right)$ + $O\left( \frac{1}{4}K \log N \right)$ + $O\left( \frac{1}{4}K \log J \right)$ & 13762s | 52580s |
| PCDL1      | $O\left( \frac{1}{4}K \log N \right)$ + $O\left( \frac{1}{4}K \log J \right)$ + $O\left( \frac{1}{4}K \log J \right)$ + $O\left( \frac{1}{4}K \log J \right)$ & $O\left( \frac{1}{4}K \log N \right)$ + $O\left( \frac{1}{4}K \log J \right)$ + $O\left( \frac{1}{4}K \log J \right)$ + $O\left( \frac{1}{4}K \log J \right)$ & 6231s | 25085s |
| PCDL2      | $O\left( \frac{1}{4}K \log N \right)$ + $O\left( \frac{1}{4}K \log J \right)$ + $O\left( \frac{1}{4}K \log J \right)$ + $O\left( \frac{1}{4}K \log J \right)$ & $O\left( \frac{1}{4}K \log N \right)$ + $O\left( \frac{1}{4}K \log J \right)$ + $O\left( \frac{1}{4}K \log J \right)$ + $O\left( \frac{1}{4}K \log J \right)$ & 5972s | 23960s |

| Filter number | 1 shot Set1 | 4 shots Set1 | 8 shots Set1 | 12 shots Set1 | 1 shot Set2 | 4 shots Set2 | 8 shots Set2 | 12 shots Set2 |
|---------------|-------------|-------------|-------------|--------------|-------------|-------------|-------------|--------------|
| 25            | 25.00       | 25.99       | 28.35       | 30.26        | 31.16       | 31.40       |
| 50            | 25.00       | 25.99       | 28.35       | 30.26        | 31.16       | 31.40       |
The reconstruction PSNRs averaged over the two test sets are listed in Table 2. First, the reconstruction results of Set2 under each measurement setting do not show obvious inferior performances compared with Set1, and even better with 4 shots. This validates that CDL can retrieve an image prior capable of generalization. Second, on all trials conducted, using 50 filters almost yields a slightly worse performance than using 25 filters. This maybe discloses that on one hand, 25 filters have been able to successfully learn the HSI intrinsic features, and on the other, using more filters means more unknown coefficient maps to estimate and thus affect the final reconstruction results. We will use the 25-filter dictionary in the sequel experiments for further exploration.

2) ROBUSTNESS TO MEASUREMENT NOISE
This experiment examines the proposed algorithm’s robustness to measurement noise. We add Gaussian noise to the previous test sets for four SNR levels, ranging from 10dB to 40dB.

Fig.6 shows the average relative MSE in dB versus the SNR. First, reconstruction from single shot is obviously inaccessible due to the sampling rate as low as 1/31. And in the 10dB SNR case that is extremely noisy, the proposed algorithm failed to recover signals from equally strong noise, with reconstruction quality deserving the shot numbers. Starting from 20dB SNR, however, the error curves show clearly the desirable behavior between SNR and MSE. The error reductions of the two groups of curves, four shots vs. eight shots and eight shots vs. twelve shots, keep an about constant ratio, 2:1, which corresponds to the improvement by accordingly increasing the number of measurements.

3) RECONSTRUCTION RESULTS OF LARGE HSI
In this experiment, we test the reconstruction performance with the 16 test images shown in Fig.5, but with higher spatial resolution 512×512. Since Fig.6 suggests a desirable result when using 12 shots with 30dB SNR, in Table 3 we collect the PSNR results under this experimental setting and compare them with the performance of CACCT (Coded Aperture Compressive Chromo-tomography) using an analytic sparsifying basis [28]. It is shown that the proposed method outperforms CACCT consistently.

Fig.7 shows the reconstruction results of four hyperspectral images from the Harvard dataset [63]. Compared with CACCT, our method exhibits higher performance in terms of HF component recovery, owing to the fusion of the learned CSC hyperspectral prior with the proposed optimization strategy.

### E. FEATURE REDISCOVERING PERFORMANCE
The last experiment is designed to examine two aspects of CDL: the information preserving capability and the feature rediscovering capability. We first conduct a CDL task using 32 HSI samples of size 128×128×31, yielding a dictionary.
$D$ consisting of 25 kernels of size $11 \times 11 \times 31$. Then the resultant feature maps as well as the learned dictionary $D$ are used to generate a new training set, which should be similar with the original one if the CDL algorithm only incurs minute information loss. And finally, a new round of CDL is started to train a new dictionary $A$, whose kernels should resemble the ones in $D$ if the CDL algorithm has reliable performance of global convergence.

We use the following atom coherence to measure the similarity of kernels in $A$ and $D$, defined as:

$$C(i, j) \triangleq \frac{\|A(:, i)^T D(:, j)\|}{\|A(:, i)\| \|D(:, j)\|},$$

where $A(:, i)$ is the vectorized version of the $i^{th}$ kernel in $A$. Note that $\max_{i,j} C(i,j)$ is the mutual coherence widely used in the compressed sensing community [64].

A pseudo-color plot of $C$ is shown in Fig. 8. The average coherence value of the first 25 biggest $C(i,j)$ is 0.6814, with...
the mutual coherence between $A$ and $D$ is 0.9680. These statistics reflect that the atoms in $A$ and $D$ have a relatively high ensemble similarity, while the difference between them is probably due to the non-convex nature of the CDL problem, even though identical initialization has been fed into the two trials of the CDL algorithm.

VII. CONCLUSION

In this paper, we cast the convolutional sparse coding and filter updating problems within the PADMM framework, leading to efficient and convergent numerical solutions to CDL for HSI. Two efficient construction methods of diagonal preconditioners have been proposed, which contributes to a Hessian free CDL algorithm for HSI with substantial algorithm acceleration and memory savings. Furthermore, we also employ PADMM to solve the CCT problem, which allows for the reconstruction of high dimensional HSI. Thereafter, we experimentally examined the usability of the learned convolutional dictionary in the CCT reconstruction problem. Numerical experiments validate that the proposed 3D CSC model is effective and generalizable in representing the HF components of HSI. The CCT reconstruction comparison results demonstrate that the CSC prior outperforms the analytic sparsifying transforms, which is maybe of interest for other compressive hyperspectral imaging schemes. Future work will examine the proposed method in real world applications and exploit learning-analytic hybrid dictionary for CCT reconstruction. Additionally, with appropriate algebraic structure adaptation, the proposed preconditioning technique can be extended to train convolutional dictionaries for other multi-dimensional signals, such as video feature learning and light field feature learning [29]. Another interesting direction is to explore the possibility that uses CSC as a sparse channel representation in wireless communication [65].

REFERENCES

[1] A. S. Charles, B. A. Olshausen, and C. J. Rozell, “Learning sparse codes for hyperspectral imagery,” IEEE J. Sel. Topics Signal Process., vol. 5, no. 5, pp. 963–978, Sep. 2011, doi: 10.1109/JSTSP.2011.2149497.

[2] L. Fang, S. Li, X. Kang, and J. A. Benediktsson, “Spectral-spatial classification of hyperspectral images with a superpixel-based discriminative sparse model,” IEEE Trans. Geosci. Remote Sens., vol. 53, no. 8, pp. 4186–4201, Aug. 2015, doi: 10.1109/TGRS.2015.2392755.

[3] X. Sun, N. M. Nasrabadi, and T. D. Tran, “Task-driven dictionary learning for hyperspectral image classification with structured sparsity constraints,” IEEE Trans. Geosci. Remote Sens., vol. 53, no. 8, pp. 4457–4471, Aug. 2015, doi: 10.1109/TGRS.2015.2399978.

[4] Y.-Q. Zhao and J. Yang, “Hyperspectral image denoising via sparse representation and low-rank constraint,” IEEE Trans. Geosci. Remote Sens., vol. 51, no. 1, pp. 296–308, Jan. 2015, doi: 10.1109/TGRS.2014.2321557.

[5] R. Zhao, B. Du, and L. Zhang, “Hyperspectral anomaly detection via a sparsity score estimation framework,” IEEE Trans. Geosci. Remote Sens., vol. 55, no. 6, pp. 3208–3222, Jun. 2017, doi: 10.1109/TGRS.2017.2664658.

[6] W. Dong, F. Fu, G. Shi, X. Cao, J. Wu, G. Li, and X. Li, “Hyperspectral image super-resolution via non-negative structured sparse representation,” IEEE Trans. Image Process., vol. 25, no. 5, pp. 2337–2352, May 2016.

[7] X.-H. Han, B. Shi, and Y. Zheng, “Self-similarity constrained sparse representation for hyperspectral image super-resolution,” IEEE Trans. Image Process., vol. 27, no. 11, pp. 5625–5637, Nov. 2018.

[8] Q. Wei, J. Bioucas-Dias, N. Dobigeon, and J.-Y. Tourneret, “Hyperspectral and multispectral image fusion based on a sparse representation,” IEEE Trans. Geosci. Remote Sens., vol. 53, no. 7, pp. 3658–3668, Jul. 2015, doi: 10.1109/TGRS.2014.2381272.

[9] L. Zhang, W. Wei, Y. Zhang, C. Shen, A. van den Hengel, and Q. Shi, “Dictionary learning for promoting structured sparsity in hyperspectral compressive sensing,” IEEE Trans. Geosci. Remote Sens., vol. 54, no. 12, pp. 7223–7235, Dec. 2016, doi: 10.1109/TGRS.2016.2598577.

[10] L. Wang, Z. Xiong, H. Huang, G. Shi, F. Wu, and W. Zeng, “High-speed hyperspectral video acquisition by combining Nyquist and compressive sampling,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 41, no. 4, pp. 857–870, Mar. 2019, doi: 10.1109/TPAMI.2018.2817496.

[11] S. G. Mallat, “A theory for multiresolution signal decomposition: The wavelet representation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 11, no. 7, pp. 674–693, Jul. 1989.

[12] M. N. Do and M. Vetterli, “The contourlet transform: An efficient directional multiresolution image representation,” IEEE Trans. Image Process., vol. 14, no. 12, pp. 2120–2145, Dec. 2005.

[13] M. Elad, M. A. T. Figueiredo, and Y. Ma, “On the role of sparse and redundant representations in image processing,” IEEE J. Sel. Topics Signal Process., vol. 1, no. 3, pp. 629–668, Sep. 2007.

[14] Z. Zhu, H. Yin, Y. Chai, Y. Li, and G. Qi, “A novel multi-modality image fusion method based on image decomposition and sparse representation,” Inf. Sci., vol. 432, pp. 516–529, Mar. 2018.

[15] R. Rubinstein, M. Zibulevsky, and M. Elad, “Double sparsity: Learning sparse dictionaries for sparse signal approximation,” IEEE Trans. Signal Process., vol. 58, no. 3, pp. 1553–1564, Mar. 2010, doi: 10.1109/TSP.2009.2033477.

[16] B. Ophir, M. Lustig, and M. Elad, “Multi-scale dictionary learning using wavelets,” IEEE J. Sel. Topics Signal Process., vol. 5, no. 5, pp. 1014–1024, Sep. 2011, doi: 10.1109/JSTSP.2011.2155032.

[17] H. Schutze, E. Barth, and T. Martinez, “Learning efficient data representations with orthogonal sparse coding,” IEEE Trans. Comput. Imag., vol. 2, no. 3, pp. 177–189, Sep. 2016, doi: 10.1109/TCI.2016.2557065.

[18] J. Sulam, B. Ophir, M. Zibulevsky, and M. Elad, “Trainlets: Dictionary learning in high dimensions,” IEEE Trans. Signal Process., vol. 64, no. 12, pp. 3180–3193, Jun. 2016, doi: 10.1109/TSP.2016.2540599.

[19] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, “Deconvolutional networks,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2010, pp. 2528–2535, doi: 10.1109/CVPR.2010.5595957.

[20] B. Hristov, A. Eriksson, and S. Lucey, “Fast convolutional sparse coding,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2013, pp. 391–398, doi: 10.1109/CVPR.2013.57.

[21] F. Heide, W. Heidrich, and G. Wetzstein, “Fast and flexible convolutional sparse coding,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2015, pp. 5135–5143, doi: 10.1109/CVPR.2015.7299149.

[22] Y. Fu, A. Lam, I. Sato, and Y. Sato, “Adaptive spatial-spectral dictionary learning for hyperspectral image restoration,” Int. J. Comput. Vis., vol. 122, no. 2, pp. 228–245, 2017.

[23] K. Bredies and H. Sun, “A proximal point algorithm of the preconditioned alternating direction method of multipliers,” J. Optim. Theory Appl., vol. 173, no. 3, pp. 878–907, Jun. 2017, doi: 10.1007/s10957-017-1112-5.

[24] Y. Jiao, Q. Jin, X. Lu, and W. Wang, “Preconditioned alternating direction method of multipliers for inverse problems with constraints,” Inverse Problems, vol. 33, no. 2, Feb. 2017, Art. no. 025004, doi: 10.1088/1364-5102/33/2/025004.

[25] J. Zhang, J. Jiang, A. Ming, X. Kang, and G. R. Arce, “Spatio-spectral modulation using a binary photomask for compressive chromonomography,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), May 2019, pp. 7710–7714, doi: 10.1109/ICASSP.2019.8682609.

[26] B. Choudhury, R. Swanson, F. Heide, G. Wetzstein, and W. Heidrich, “Consensus convolutional sparse coding,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 4290–4298, doi: 10.1109/ICCV.2017.459.

[27] M. Sørel and F. Sørbøe, “Fast convolutional sparse coding using matrix inversion lemma,” Digit. Signal Process., vol. 55, pp. 44–51, Aug. 2016.
