Toward Robust Hyperspectral Unmixing: Mixed Noise Modeling and Image-Domain Regularization

Kazuki Naganuma, *Graduate Student Member, IEEE*, and Shunsuke Ono, *Senior Member, IEEE*

***Abstract*—Hyperspectral (HS) unmixing is the process of decomposing an HS image into material-specific spectra (endmembers) and their spatial distributions (abundance maps). Existing unmixing methods have two limitations with respect to noise robustness. First, if the input HS image is highly noisy, even if the balance between sparse and piecewise-smooth regularizations for abundance maps is carefully adjusted, noise may remain in the estimated abundance maps or undesirable artifacts may appear. Second, existing methods do not explicitly account for the effects of stripe noise, which is common in HS measurements, in their formulations, resulting in significant degradation of unmixing performance when such noise is present in the input HS image. To overcome these limitations, we propose a new robust HS unmixing method based on constrained convex optimization. Our method employs, in addition to the two regularizations for the abundance maps, regularizations for the HS image reconstructed by mixing the estimated abundance maps and endmembers. This strategy makes the unmixing process much more robust in highly noisy scenarios, under the assumption that the abundance maps used to reconstruct the HS image with desirable spatio-spectral structure are also expected to have desirable properties. Furthermore, our method is designed to accommodate a wider variety of noise including stripe noise. To solve the formulated optimization problem, we develop an efficient algorithm based on a preconditioned primal-dual splitting method, which can automatically determine appropriate stepsizes based on the problem structure. Experiments on synthetic and real HS images demonstrate the advantages of our method over existing methods.**

***Index Terms*—Constrained optimization, hyperspectral (HS) unmixing, mixed noise, primal-dual splitting, stripe noise.**

I. Introduction

HYPERSPECTRAL (HS) images are 3-D cube data consisting of 2-D spatial and 1-D spectral information. Compared to grayscale or RGB images, HS images provide more than several hundred bands, each of which contains specific unique wavelength characteristics of materials, such as minerals, soils, and liquids. Therefore, HS images have various applications, such as ecology, mineralogy, biotechnology, and agriculture [1], [2], [3], [4]. Due to the tradeoff between spatial resolution and spectral resolution, HS sensors do not have a sufficient spatial resolution, resulting in containing multiple components (called endmembers) in a pixel [5], which is referred to as a mixel. The process of decomposing the mixel into endmembers and their abundance maps is called unmixing. Unmixing has been actively studied in the remote sensing field because it is essential for HS image analysis [6], [7] and other applications, such as denoising [8], [9] and data fusion [10], [11].

Unmixing methods fall into two categories according to their assumptions: 1) nonblind [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31] and 2) blind unmixing [32], [33], [34], [35], [36]. Nonblind unmixing methods estimate abundance maps from a given endmember library. Endmembers in the library are potentially much larger in number than endmembers included in real HS images, i.e., its corresponding abundance maps become sparse. On the other hand, blind unmixing methods simultaneously estimate an endmember library and abundance maps, allowing us to obtain the abundances of endmembers whose spectral libraries are unknown.

For blind unmixing, nonnegative matrix factorization-based approaches [32], [33], [34] and learning-based approaches [35], [36], [37] have attracted attention. Nonnegative matrix factorization-based methods design and solve an optimization problem that incorporates the functions of the product of an endmember matrix and an abundance map matrix. When solving the optimization problem, they take an approach that iterates alternate updates of the two matrices: updating the endmember library matrix by solving the subproblem with the abundance map matrix fixed, updating the abundance maps by solving the subproblem with the endmember library matrix fixed using some nonblind unmixing method. Learning-based methods often involve the following steps: Extraction of initial endmembers from an input HS image, estimation of corresponding initial abundance maps by some nonblind unmixing methods, and then learning of sophisticated unmixing and reconstruction networks based on this information. Therefore, nonblind unmixing is a fundamental task that must precede blind unmixing. Henceforth, nonblind unmixing will simply be referred to as unmixing.

Although very accurate unmixing can be achieved using state-of-the-art methods if a noise-free HS image is available, real-world HS images are often contaminated by various types of noise, such as Gaussian noise, outliers, missing values, and stripe noise due to environmental factors and sensor failures. Such noise obviously has a negative impact on unmixing performance

Manuscript received 6 October 2023; revised 24 January 2024; accepted 16 March 2024. Date of publication 20 March 2024; date of current version 17 April 2024. This work was supported in part by JST ACT-X under Grant JPMJAX23CJ, in part by JST PRESTO under Grant JPMJPR21C4, in part by JST AdCORP under Grant JPMJKB2307, in part by JSPS KAKENHI under Grant 22H03610, Grant 22H00512, and Grant 23H01415, and in part by the Grant-in-Aid for JSPS Fellows under Grant 23KJ0912. *(Corresponding author: Kazuki Naganuma.)*

The authors are with the Department of Computer Science, Tokyo Institute of Technology, Yokohama 226-8503, Japan (e-mail: naganuma.k.aa@m.titech.ac.jp; ono@c.titech.ac.jp).

Digital Object Identifier 10.1109/JSTARS.2024.3379558

and needs to be dealt with appropriately. The simplest way is a two-step approach, where noise is first removed from a given HS image beforehand, followed by unmixing. However, such methods are also likely to remove even important spectral information. It is therefore essential to develop a method that can *simultaneously* separate noise (without affecting spectral information) during the unmixing process, which we refer to as *noise-robust* unmixing.

Many noise-robust unmixing techniques explicitly model noises and then take the approach of solving optimization problems that incorporate functions characterizing abundance maps. Based on the fact that HS images consist of a small fraction of the endmembers in a library, the methods in [12], [13], [14], [15], [16], and [17] employ a sparse regularization. Abundance maps are also piecewise smooth because neighboring pixels often have the same endmembers. To capture the nature, the methods in [18], [19], [20], [21], and [22] adopt a combination of sparse and piecewise-smooth regularizations. As a more advanced approach to promote the sparsity of abundance maps, some methods adopt a coarse abundance map-based weighted sparse unmixing approach [23], [24], [25], which includes the following three steps. First, this approach segments a target HS image into superpixel blocks and averages the pixels of the superpixel blocks to obtain a coarsely denoised HS image. Then, by applying a sparse unmixing method (e.g., the method in [12]) to the coarse HS image, coarse abundance maps are generated. Finally, based on the coarse abundance maps, superpixelwise and endmemberwise weights are computed to promote the weighted sparsity of abundance maps. In addition, the methods in [26], [27], and [28] estimate abundance maps using a regularization based on deep neural networks, and the methods in [29], [30], and [31] adopt a combination of sparse and low-rank regularizations.

As we have discussed, various studies have been carried out to mitigate the effects of noise in unmixing, but there are still two limitations in terms of robustness to noise. The first is that the performance of unmixing is severely degraded when the input HS image is contaminated with high levels of noise. The second is that existing unmixing methods cannot adequately deal with stripe noise.

As reviewed in the previous subsection, many existing unmixing methods use a combination of sparse and piecewise-smooth regularization to characterize the abundance maps. However, as shown in Fig. 1, balancing these regularizations becomes very difficult when unmixing HS images contaminated with high levels of noise. In fact, if the weight of the sparse regularization is increased, a large amount of noise remains in the estimated abundance maps. Conversely, if the weight of the piecewise-smooth regularization is increased, the estimated abundance maps will contain many inappropriate components that are not present in the original HS image. In existing methods, adjusting the weights to avoid both problems is a very sensitive and tedious task.

To resolve this difficulty, we focus on the regularizations for the HS image reconstructed by mixing the estimated abundance maps and the endmembers, which we call image-domain regularizations, in addition to the regularizations for the abundance maps. Our assumption is the following: if the reconstructed HS image has desirable properties in its spatio-spectral structure,

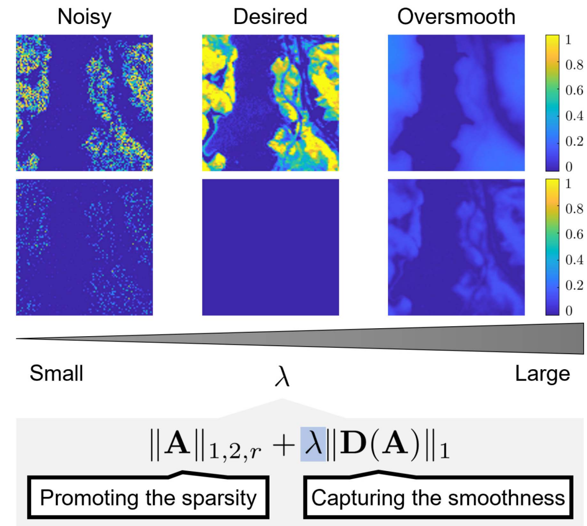

Fig. 1. Difficulty in dealing with high-level noise in unmixing.

then the estimated abundance maps used for reconstruction should also have desirable properties. Therefore, we believe that incorporating spatio-spectral regularization for HS images into the unmixing formulation can improve the unmixing performance in high-noise situations where abundance maps are difficult to estimate using existing methods. Fortunately, in the context of HS image restoration, many effective HS image regularizations have been studied [38], [39], [40], [41], [42], [43], [44]. By adopting them as image-domain regularizations, we can robustify the unmixing process under highly noisy scenarios.

Regarding the second limitation in dealing with stripe noise, existing unmixing methods mainly deal with Gaussian noise and sparse noise. However, in addition to these noises, actual HS images are often contaminated with stripe noise, mainly due to external disturbances and calibration errors [45], [46], [47]. Since stripe noise is not Gaussian and is often not sparse [46], it cannot be handled by existing methods, leading to performance degradation in unmixing.

Based on the above discussion, this article proposes robust HS unmixing using image-domain regularization (RHUIDR). We formulate the unmixing problem as a constrained convex optimization problem. In order to solve the optimization problem, we develop an efficient algorithm based on the preconditioned primal-dual splitting method (P-PDS) [48] with an operator-norm-based stepsize selection method [49]. In terms of the features of RHUIDR, the contributions of this article can be summarized as follows.

1) *Robust to High Levels of Noise:* RHUIDR employs not only the abundance map regularizations but also image-domain regularizations, which robustify the unmixing process under highly noisy scenarios.
2) *Robust to Mixed Noise Including Stripe Noise):* By explicitly modeling three types of noise (Gaussian noise, sparse noise, and stripe noise), as in (14), RHUIDR can adequately handle mixed noise, including stripe noise, which is difficult to handle in existing methods.
3) *Easy to Adjust Hyperparameters:* In the formulated optimization problem, we model data-fidelity and noise terms

as hard constraints instead of adding them to the objective function. This type of constrained formulation decouples interdependent hyperparameters into independent ones, thus facilitating parameter settings, which will be detailed in Section III-A.

4) *Avoiding Adjusting Stepsizes:* Unlike the optimization algorithms used in existing unmixing methods, our P-PDS-based algorithm can automatically determine the appropriate stepsizes based on the problem structure.

Experiments on synthetic and real HS images demonstrate the advantages of RHUIDR over existing methods.

The rest of this article is organized as follows. In Section II, we introduce mathematical tools. In Section III, we explain the proposed method, RHUIDR, with its formulation and algorithm. In Section IV, we conduct experiments to show the superiority of RHUIDR over the existing methods. Finally, Section V concludes this article.

II. Preliminaries

A. Notations

In this article, we denote the sets of real numbers and nonnegative real numbers as $\mathbb{R}$ and $\mathbb{R}_+$, respectively. Matrices are denoted by capitalized boldface letters (e.g., $\mathbf{X}$), and the element at the ith row and hte jth column of matrix $\mathbf{X}$ is denoted by $X_{i,j}$ or $[\mathbf{X}]_{i,j}$. An HS image with the number of bands l and spatial size $n_1 \times n_2$ is treated as a matrix $\mathbf{H} \in \mathbb{R}^{l \times n_1 n_2}$ of size $l \times n_1 n_2$ and $[\mathcal{H}]_{i,j,k}$ indicates the (i,j,k)th value of the cube data $\mathcal{H} \in \mathbb{R}^{n_1 \times n_2 \times l}$ corresponding to $\mathbf{H}$. The ℓ_1-norm $\|\cdot\|_1$, the Frobenius norm $\|\cdot\|_F$, the mixed $\ell_{1,2}$-norm grouped by row $\|\cdot\|_{1,2,r}$, and the mixed $\ell_{1,2}$-norm grouped by column $\|\cdot\|_{1,2,c}$ are defined by $\|\mathbf{X}\|_1 = \sum_{i,j} |X_{i,j}|$, $\|\mathbf{X}\|_F = \sqrt{\sum_{i,j} X_{i,j}^2}$, $\|\mathbf{X}\|_{1,2,r} = \sum_i \sqrt{\sum_j X_{i,j}^2}$, and $\|\mathbf{X}\|_{1,2,c} = \sum_j \sqrt{\sum_i X_{i,j}^2}$, respectively. Let $\mathbf{G} : \mathbb{R}^{M_1 \times N_1} \to \mathbb{R}^{M_2 \times N_2}$ be a linear operator. A linear operator $\mathbf{G}^* : \mathbb{R}^{M_2 \times N_2} \to \mathbb{R}^{M_1 \times N_1}$ is called the adjoint operator of $\mathbf{G}$ if it is satisfied with $\langle \mathbf{G}(\mathbf{X}), \mathbf{Y} \rangle = \langle \mathbf{X}, \mathbf{G}^*(\mathbf{Y}) \rangle$ for any $\mathbf{X} \in \mathbb{R}^{M_1 \times N_1}$ and $\mathbf{Y} \in \mathbb{R}^{M_2 \times N_2}$.

B. Preconditioned Primal-Dual Splitting Method (P-PDS)

Let $f_1, \ldots, f_N, g_1, \ldots, g_M$ be proximable[1] proper lower semicontinuous convex functions. Consider a convex optimization problem of the following form:

$$\min_{\substack{\mathbf{Y}_1,\ldots,\mathbf{Y}_N,\\ \mathbf{Z}_1,\ldots,\mathbf{Z}_M}} \sum_{i=1}^{N} f_i(\mathbf{Y}_i) + \sum_{j=1}^{M} g_j(\mathbf{Z}_j)$$
$$\text{s.t.} \begin{cases} \mathbf{Z}_i = \sum_{i=1}^{N} \mathbf{G}_{1,i}(\mathbf{Y}_i) \\ \vdots \\ \mathbf{Z}_M = \sum_{i=1}^{N} \mathbf{G}_{M,i}(\mathbf{Y}_i) \end{cases} \tag{1}$$

where $\mathbf{G}_{j,i}$ $(i = 1, \ldots, N, j = 1, \ldots, M)$ are linear operators. We define the *proximity operator* of f_i (and g_j as well) with a parameter $\gamma > 0$ by

$$\mathrm{prox}_{\gamma f_i}(\mathbf{X}) := \operatorname*{argmin}_{\mathbf{Y} \in \mathbb{R}^{M \times N}} f_i(\mathbf{Y}) + \frac{1}{2\gamma} \|\mathbf{X} - \mathbf{Y}\|_F^2. \tag{2}$$

Then, the P-PDS solves Problem (1) by the following iterative procedures:

$$\left\lfloor \begin{array}{ll} \widetilde{\mathbf{Y}}_1 & \leftarrow \mathbf{Y}_1^{(t)} - \gamma_{1,1} \sum_{j=1}^{M} \mathbf{G}_{j,1}^*(\mathbf{Z}_j^{(t)}) \\ \mathbf{Y}_1^{(t+1)} & \leftarrow \mathrm{prox}_{\gamma_{1,1} f_1}(\widetilde{\mathbf{Y}}_1) \\ & \vdots \\ \widetilde{\mathbf{Y}}_N & \leftarrow \mathbf{Y}_N^{(t)} - \gamma_{1,N} \sum_{j=1}^{M} \mathbf{G}_{j,N}^*(\mathbf{Z}_j^{(t)}) \\ \mathbf{Y}_N^{(t+1)} & \leftarrow \mathrm{prox}_{\gamma_{1,N} f_N}(\widetilde{\mathbf{Y}}_N) \\ \widetilde{\mathbf{Z}}_1 & \leftarrow \mathbf{Z}_1^{(t)} + \gamma_{2,1} \sum_{i=1}^{N} \mathbf{G}_{1,i}(2\mathbf{Y}_i^{(t+1)} - \mathbf{Y}_i^{(t)}) \\ \mathbf{Z}_1^{(t+1)} & \leftarrow \widetilde{\mathbf{Z}}_1 - \gamma_{2,1} \mathrm{prox}_{\frac{1}{\gamma_{2,1}} g_1}(\frac{1}{\gamma_{2,1}} \widetilde{\mathbf{Z}}_1) \\ & \vdots \\ \widetilde{\mathbf{Z}}_M & \leftarrow \mathbf{Z}_M^{(t)} + \gamma_{2,M} \sum_{i=1}^{N} \mathbf{G}_{M,i}(2\mathbf{Y}_i^{(t+1)} - \mathbf{Y}_i^{(t)}) \\ \mathbf{Z}_M^{(t+1)} & \leftarrow \mathbf{Z}_M^{(t)} - \gamma_{2,M} \mathrm{prox}_{\frac{1}{\gamma_{2,M}} g_M}(\frac{1}{\gamma_{2,M}} \widetilde{\mathbf{Z}}_M) \end{array} \right. \tag{3}$$

where $\gamma_{1,i}$ $(i = 1, \ldots, N)$ and $\gamma_{2,j}$ $(j = 1, \ldots, M)$ are the stepsize parameters. The stepsize parameters can be determined automatically as follows [49]:

$$\gamma_{1,i} = \frac{1}{\sum_{j=1}^{M} \|\mathbf{G}_{j,i}\|_{\mathrm{op}}^2}, \ \gamma_{2,j} = \frac{1}{N} \tag{4}$$

where $\|\mathbf{G}_{j,i}\|_{\mathrm{op}}^2$ is the operator norm[2] of $\mathbf{G}_{j,i}$. However, the operator norms of some linear operators cannot be easily calculated (e.g., difference operators and composite operators of linear operators). Therefore, we can use their upper bounds $\mu_{j,i} \in [\|\mathbf{G}_{j,i}\|_{\mathrm{op}}, \infty)$ to determine the stepsize parameters as

$$\gamma_{1,i} = \frac{1}{\sum_{j=1}^{M} \mu_{j,i}^2}, \ \gamma_{2,j} = \frac{1}{N}. \tag{5}$$

From [48, Th. 1] and [49, Th. III.2], the sequences generated by P-PDS (3) with the stepsizes in (5) are guaranteed to converge to a solution of Problem (1).

C. Regularizations for an HS Image

This section introduces the regularizations for an HS image $\mathbf{H} \in \mathbb{R}^{l \times n_1 n_2}$. Let $\mathbf{D}_v : \mathbb{R}^{l \times n_1 n_2} \to \mathbb{R}^{l \times n_1 n_2}$, $\mathbf{D}_h : \mathbb{R}^{l \times n_1 n_2} \to \mathbb{R}^{l \times n_1 n_2}$, and $\mathbf{D}_b : \mathbb{R}^{l \times n_1 n_2} \to \mathbb{R}^{l \times n_1 n_2}$ be, respectively, vertical, horizontal, and spectral difference operators, which are given by

$$[\mathcal{D}_v]_{i,j,k} := \begin{cases} [\mathcal{H}]_{i+1,j,k} - [\mathcal{H}]_{i,j,k}, & \text{if } i < n_1 \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

$$[\mathcal{D}_h]_{i,j,k} := \begin{cases} [\mathcal{H}]_{i,j+1,k} - [\mathcal{H}]_{i,j,k}, & \text{if } j < n_2 \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

[1]If an efficient computation of the proximity operator of f is available, we call f proximable.

[2]Let $\mathbf{G} : \mathbb{R}^{n_1, m_1} \to \mathbb{R}^{n_2, m_2}$ be a linear operator. Then, the operator norm of $\mathbf{G}$ is defined by $\|\mathbf{G}\|_{\mathrm{op}} := \sup_{\mathbf{X} \neq \mathbf{O}} \|\mathbf{G}(\mathbf{X})\|_F / \|\mathbf{X}\|_F$.

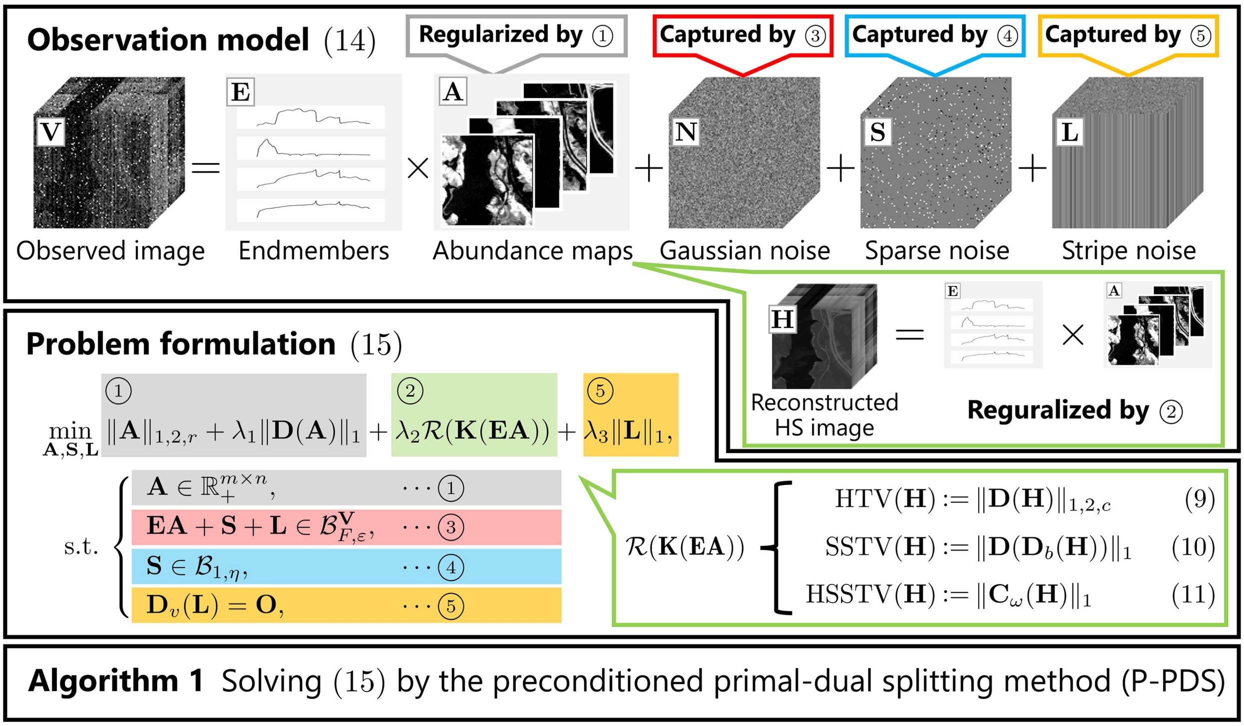

Fig. 2. Illustration of the proposed method, i.e., RHUIDR.

$$[\mathcal{D}_b]_{i,j,k} := \begin{cases} [\mathcal{H}]_{i,j,k+1} - [\mathcal{H}]_{i,j,k}, & \text{if } k < l \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

where $\mathcal{H} \in \mathbb{R}^{n_1\times n_2\times l}$, $\mathcal{D}_v \in \mathbb{R}^{n_1\times n_2\times l}$, $\mathcal{D}_h \in \mathbb{R}^{n_1\times n_2\times l}$, and $\mathcal{D}_b \in \mathbb{R}^{n_1\times n_2\times l}$ are the 3-D data corresponding to $\mathbf{H}$, $\mathbf{D}_v(\mathbf{H})$, $\mathbf{D}_h(\mathbf{H})$, and $\mathbf{D}_b(\mathbf{H})$, respectively. Then, HTV [38], SSTV [41], and HSSTV [42] are defined by

$$\mathrm{HTV}(\mathbf{H}) := \|\mathbf{D}(\mathbf{H})\|_{1,2,c} \tag{9}$$

$$\mathrm{SSTV}(\mathbf{H}) := \|\mathbf{D}(\mathbf{D}_b(\mathbf{H}))\|_1 \tag{10}$$

$$\mathrm{HSSTV}(\mathbf{H}) := \|\mathbf{C}_\omega(\mathbf{H})\|_1 \tag{11}$$

where $\mathbf{D}$ is the spatial difference operator

$$\mathbf{D}(\mathbf{H}) := \begin{bmatrix} \mathbf{D}_v(\mathbf{H}) \\ \mathbf{D}_h(\mathbf{H}) \end{bmatrix} \tag{12}$$

and $\mathbf{C}_\omega$ is a combination of spatial and spatio-spectral difference operators with a balancing parameter $\omega > 0$

$$\mathbf{C}_\omega(\mathbf{H}) := \begin{bmatrix} \mathbf{D}(\mathbf{D}_b(\mathbf{H})) \\ \omega\mathbf{D}(\mathbf{H}) \end{bmatrix}. \tag{13}$$

HTV captures spectral correlations by promoting the sparsity of spatial differences grouped by the spectral direction. SSTV captures piecewise smoothness in the spatial and spectral directions by using the composite operator of the spatial and spectral differences (spatio-spectral difference). However, it does not sufficiently evaluate direct spatial piecewise smoothness, resulting in residual noise and artifacts. HSSTV promotes both spatio-spectral and direct spatial smoothness, and thus, is a more powerful regularization in general.

TABLE I
SPECIFIC FUNCTION $\mathcal{R}$ AND LINEAR OPERATOR $\mathbf{K}$ IN EACH RCONSTRUCTED-IMAGE REGULARIZATION

Regularizations	$\mathcal{R}$	$\mathbf{K}$
HTV (9)	$\|\cdot\|_{1,2,c}$	$\mathbf{D}$
SSTV (10)	$\|\cdot\|_1$	$\mathbf{D}\circ\mathbf{D}_b$
HSSTV (11)	$\|\cdot\|_1$	$\mathbf{C}_\omega$

TABLE II
STEPSIZES $\gamma_1, \gamma_2, \gamma_3$, AND γ_4 FOR EACH ALGORITHM THAT SOLVES AN OPTIMIZATION PROBLEM INCORPRATING EACH RECONSTRUCTED-IMAGE REGULARIZATION

Regularizations	γ_1	γ_2	γ_3	γ_4
HTV (9)	$\frac{1}{9+9\sigma_1(\mathbf{A})^2}$	1	$\frac{1}{5}$	$\frac{1}{3}$
SSTV (10)	$\frac{1}{9+33\sigma_1(\mathbf{A})^2}$	1	$\frac{1}{5}$	$\frac{1}{3}$
HSSTV (11)	$\frac{1}{9+(33+8\omega^2)\sigma_1(\mathbf{A})^2}$	1	$\frac{1}{5}$	$\frac{1}{3}$

III. PROPOSED METHOD

A general diagram of the proposed method, RHUIDR, is shown in Fig. 2. In the following, we first introduce an observation model with three types of noise. Based on the model, we then formulate the unmixing problem as a constrained convex optimization problem. Finally, we describe a P-PDS-based algorithm to efficiently solve the optimization problem with its computational complexity.

Algorithm 1: A P-PDS-Based Algorithm for Solving Prob. (15).

Input: $\mathbf{V}$, $\mathbf{E}$, λ_1, λ_2, λ_3, ε, and η
Output: $\mathbf{A}^{(t)}$, $\mathbf{S}^{(t)}$, $\mathbf{L}^{(t)}$

Initialize $\mathbf{A}^{(0)}$, $\mathbf{S}^{(0)}$, $\mathbf{L}^{(0)}$, $\mathbf{Z}_1^{(0)}$, $\mathbf{Z}_2^{(0)}$, $\mathbf{Z}_3^{(0)}$, $\mathbf{Z}_4^{(0)}$, and $\mathbf{Z}_5^{(0)}$;
Set $\gamma_1, \gamma_2, \gamma_3$, and γ_4 as in Table II;
while until a stopping criterion is satisfied **do**
$\quad \widetilde{\mathbf{A}} \leftarrow \mathbf{Z}_1^{(t)} + \mathbf{D}^*(\mathbf{Z}_2^{(t)}) + \mathbf{E}^*(\mathbf{K}^*(\mathbf{Z}_3^{(t)})) + \mathbf{E}^*\mathbf{Z}_4^{(t)}$;
$\quad \widetilde{\mathbf{A}} \leftarrow \mathbf{A}^{(t)} - \gamma_1 \widetilde{\mathbf{A}}$;
$\quad \mathbf{A}^{(t+1)} \leftarrow \mathrm{prox}_{\gamma_1 \iota_{\mathbb{R}_+^{m\times n}}}(\widetilde{\mathbf{A}})$ by (20);
$\quad \widetilde{\mathbf{S}} \leftarrow \mathbf{S}^{(t)} - \gamma_2 \mathbf{Z}_4^{(t)}$;
$\quad \mathbf{S}^{(t+1)} \leftarrow \mathrm{prox}_{\gamma_2 \iota_{\mathcal{B}_{1,\eta}}}(\widetilde{\mathbf{S}})$;
$\quad \widetilde{\mathbf{L}} \leftarrow \mathbf{L}^{(t)} - \gamma_3(\mathbf{Z}_4^{(t)} + \mathbf{D}_v^*(\mathbf{Z}_5^{(t)}))$;
$\quad \mathbf{L}^{(t+1)} \leftarrow \mathrm{prox}_{\gamma_3\lambda_3\|\cdot\|_1}(\widetilde{\mathbf{L}})$ by (21);
$\quad \widetilde{\mathbf{Z}}_1 \leftarrow \mathbf{Z}_1^{(t)} + \gamma_4(2\mathbf{A}^{(t+1)} - \mathbf{A}^{(t)})$;
$\quad \mathbf{Z}_1^{(t+1)} \leftarrow \widetilde{\mathbf{Z}}_1 - \gamma_4 \mathrm{prox}_{\frac{1}{\gamma_4}\|\cdot\|_{1,2,r}}(\frac{\widetilde{\mathbf{Z}}_1}{\gamma_4})$ by (22);
$\quad \widetilde{\mathbf{Z}}_2 \leftarrow \mathbf{Z}_2^{(t)} + \gamma_4 \mathbf{D}(2\mathbf{A}^{(t+1)} - \mathbf{A}^{(t)})$;
$\quad \mathbf{Z}_2^{(t+1)} \leftarrow \widetilde{\mathbf{Z}}_2 - \gamma_4 \mathrm{prox}_{\frac{\lambda_1}{\gamma_4}\|\cdot\|_1}(\frac{\widetilde{\mathbf{Z}}_2}{\gamma_4})$ by (21);
$\quad \widetilde{\mathbf{Z}}_3 \leftarrow \mathbf{Z}_3^{(t)} + \gamma_4 \mathbf{K}(\mathbf{E}(2\mathbf{A}^{(t+1)} - \mathbf{A}^{(t)}))$;
$\quad \mathbf{Z}_3^{(t+1)} \leftarrow \widetilde{\mathbf{Z}}_3 - \gamma_4 \mathrm{prox}_{\frac{\lambda_2}{\gamma_4}\mathcal{R}}(\frac{\widetilde{\mathbf{Z}}_3}{\gamma_4})$ by (21) or (23);
$\quad \widetilde{\mathbf{Z}}_4' \leftarrow 2(\mathbf{E}\mathbf{A}^{(t+1)} + \mathbf{S}^{(t+1)} + \mathbf{L}^{(t+1)})$;
$\quad \widetilde{\mathbf{Z}}_4 \leftarrow \mathbf{E}\mathbf{A}^{(t)} + \mathbf{S}^{(t)} + \mathbf{L}^{(t)}$;
$\quad \widetilde{\mathbf{Z}}_4 \leftarrow \mathbf{Z}_4^{(t)} + \gamma_4(\widetilde{\mathbf{Z}}_4' - \widetilde{\mathbf{Z}}_4)$;
$\quad \mathbf{Z}_4^{(t+1)} \leftarrow \widetilde{\mathbf{Z}}_4 - \gamma_4 \mathrm{prox}_{\frac{1}{\gamma_4}\iota_{\mathcal{B}_{F,\varepsilon}^{\mathbf{V}}}}(\frac{\widetilde{\mathbf{Z}}_4}{\gamma_4})$ by (24);
$\quad \widetilde{\mathbf{Z}}_5 \leftarrow \mathbf{Z}_5^{(t)} + \gamma_4 \mathbf{D}_v(2\mathbf{L}^{(t+1)} - \mathbf{L}^{(t)})$;
$\quad \mathbf{Z}_5^{(t+1)} \leftarrow \widetilde{\mathbf{Z}}_5 - \gamma_4 \mathrm{prox}_{\frac{1}{\gamma_4}\iota_{\{\mathbf{O}\}}}(\frac{\widetilde{\mathbf{Z}}_5}{\gamma_4})$ by (25);
$\quad t \leftarrow t + 1$;
end while

A. Problem Formulation

Let $\mathbf{E} \in \mathbb{R}^{l\times m}$, $\bar{\mathbf{A}} \in \mathbb{R}^{m\times n}$, $\bar{\mathbf{N}} \in \mathbb{R}^{l\times n}$, $\bar{\mathbf{S}} \in \mathbb{R}^{l\times n}$, and $\bar{\mathbf{L}} \in \mathbb{R}^{l\times n}$ be a given endmember library, a true abundance matrix, Gaussian noise, sparse noise, and stripe noise (need not be sparse), respectively. Consider the following observation model:

$$\mathbf{V} = \mathbf{E}\bar{\mathbf{A}} + \bar{\mathbf{N}} + \bar{\mathbf{S}} + \bar{\mathbf{L}}. \tag{14}$$

Note that this model explicitly deals with stripe noise as an additive component $\bar{\mathbf{L}}$. Based on (14), we formulate an unmixing problem as the following constrained convex optimization problem:

$$\min_{\mathbf{A},\mathbf{S},\mathbf{L}} \|\mathbf{A}\|_{1,2,r} + \lambda_1\|\mathbf{D}(\mathbf{A})\|_1 + \lambda_2\mathcal{R}(\mathbf{K}(\mathbf{E}\mathbf{A})) + \lambda_3\|\mathbf{L}\|_1$$
$$\text{s.t.} \begin{cases} \mathbf{A} \in \mathbb{R}_+^{m\times n} \\ \mathbf{E}\mathbf{A} + \mathbf{S} + \mathbf{L} \in \mathcal{B}_{F,\varepsilon}^{\mathbf{V}} \\ \mathbf{S} \in \mathcal{B}_{1,\eta} \\ \mathbf{D}_v(\mathbf{L}) \in \{\mathbf{O}\} \end{cases} \tag{15}$$

TABLE III
COMPUTATIONAL COMPLEXITIES OF EACH OPERATION

Operations	O-notation
$\mathbf{E}\mathbf{A}$, ($\mathbf{E} \in \mathbb{R}^{l\times m}$ and $\mathbf{A} \in \mathbb{R}^{m\times n}$)	$O(nml)$
$\mathbf{D}(\mathbf{A})$, ($\mathbf{A} \in \mathbb{R}^{m\times n}$)	$O(nm)$
$\mathbf{D}(\mathbf{H})$, ($\mathbf{H} \in \mathbb{R}^{l\times n}$)	$O(nl)$
$\mathbf{K}(\mathbf{H})$, $\begin{cases} \mathbf{K} = \mathbf{D},\ (\mathbf{H} \in \mathbb{R}^{l\times n}) \\ \mathbf{K} = \mathbf{D}\circ\mathbf{D}_b,\ (\mathbf{H} \in \mathbb{R}^{l\times n}) \\ \mathbf{K} = \mathbf{C}_\omega,\ (\mathbf{H} \in \mathbb{R}^{l\times n}) \end{cases}$	$O(nl)$
$\mathrm{prox}_{\gamma_1\iota_{\mathbb{R}_+^{m\times n}}}(\mathbf{A})$, ($\mathbf{A} \in \mathbb{R}^{m\times n}$)	$O(nm)$
$\mathrm{prox}_{\gamma_2\iota_{\mathcal{B}_{1,\eta}}}(\mathbf{S})$, ($\mathbf{S} \in \mathbb{R}^{l\times n}$)	$O(nl\log nl)$
$\mathrm{prox}_{\gamma_3\lambda_3\|\cdot\|_1}(\mathbf{L})$, ($\mathbf{L} \in \mathbb{R}^{l\times n}$)	$O(nl)$
$\mathrm{prox}_{\frac{1}{\gamma_4}\|\cdot\|_{1,2,r}}(\mathbf{Z}_1)$, ($\mathbf{Z}_1 \in \mathbb{R}^{m\times n}$)	$O(nm)$
$\mathrm{prox}_{\frac{\lambda_1}{\gamma_4}\|\cdot\|_1}(\mathbf{Z}_2)$, ($\mathbf{Z}_2 \in \mathbb{R}^{2m\times n}$)	$O(nm)$
$\mathrm{prox}_{\frac{\lambda_2}{\gamma_4}\mathcal{R}}(\mathbf{Z}_3)$, $\begin{cases} \mathcal{R} = \|\cdot\|_{1,2,c},\ (\mathbf{Z}_3 \in \mathbb{R}^{2l\times n}) \\ \mathcal{R} = \|\cdot\|_1,\ (\mathbf{Z}_3 \in \mathbb{R}^{2l\times n}) \\ \mathcal{R} = \|\cdot\|_1,\ (\mathbf{Z}_3 \in \mathbb{R}^{4l\times n}) \end{cases}$	$O(nl)$
$\mathrm{prox}_{\frac{1}{\gamma_4}\iota_{\mathcal{B}_{F,\varepsilon}^{\mathbf{V}}}}(\mathbf{Z}_4)$, ($\mathbf{Z}_4 \in \mathbb{R}^{l\times n}$)	$O(nl)$

where $\lambda_1 > 0$, $\lambda_2 > 0$, and $\lambda_3 > 0$ are hyperparameters that balance each term. The first term is the joint-sparse regularization that evaluates the row sparsity of abundance maps $\mathbf{A}$. The second term promotes the piecewise smoothness of $\mathbf{A}$. The first constraint guarantees the nonnegativity of $\mathbf{A}$. Note that we do not explicitly adopt the abundance sum-to-one constraint. This is because, in real-world situations, the abundance sum-to-one constraint tends to be a strong assumption for the linear mixing model based unmixing because the spectral signatures are often affected by a positive scaling factor that varies from pixel to pixel [12].

The third term is the regularization of the reconstructed HS image. This image-domain regularization enhances the noise robustness of unmixing beyond the capability of the abundance regularizations by capturing the desirable nature of the reconstructed HS image (e.g., spatio-spectral correlation). In this article, we focus on three image-domain regularizations: HTV in (9), SSTV in (10), and HSSTV in (11). In each case, $\mathcal{R}$ and $\mathbf{K}$ are defined, as shown in Table I. By further generalizing the third term, RHUIDR can incorporate other regularizations proposed, e.g., in [43] and [44].

The second constraint serves as data-fidelity to the observed HS image $\mathbf{V}$ with the Frobenius norm ball $\mathcal{B}_{F,\varepsilon}^{\mathbf{V}}$ with the center $\mathbf{V}$ and radius ε, defined by

$$\mathcal{B}_{F,\varepsilon}^{\mathbf{V}} := \{\mathbf{X} \in \mathbb{R}^{l\times n} \mid \|\mathbf{V} - \mathbf{X}\|_F \leq \varepsilon\}. \tag{16}$$

The third constraint evaluates the sparsity of $\mathbf{S}$ with the ℓ_1-norm ball $\mathcal{B}_{1,\eta}$ with center $\mathbf{O}$ and radius η, defined by

$$\mathcal{B}_{1,\eta} := \{\mathbf{X} \in \mathbb{R}^{l\times n} \mid \|\mathbf{X}\|_1 \leq \eta\}. \tag{17}$$

As described in the third contribution, using such constraints instead of data-fidelity and sparse terms makes it easy to adjust

TABLE IV
ASSUMPTIONS AND NOISE CONSIDERED IN EACH METHOD

Methods	Assumption	Noise		
		Gaussian	Sparse	Stripe
CLSUnSAL [13]	nonblind	✓	-	-
JSTV [19]	nonblind	✓	✓	-
RSSUn-TV [22]	nonblind	✓	-	-
LGSU [16]	nonblind	✓	-	-
UnDIP [28]	nonblind	✓	-	-
EGU-Net [37]	blind	✓	-	-
RDSWSU [25]	nonblind	✓	-	-
MdLRR [31]	nonblind	✓	-	-
RHUIDR (Ours)	nonblind	✓	✓	✓

hyperparameters since the parameters can be determined based only on noise intensity. Indeed, this kind of constrained formulation has played an important role in facilitating parameter setup of signal recovery problems [50], [51], [52], [53], [54]. The detailed setting of these parameters ε and η is shown in Section V-B.

The fourth term controls the intensity of stripe noise $\mathbf{L}$ and the fourth constraint captures the vertical flatness property by imposing zero to the vertical gradient of $\mathbf{L}$. The term and constraint accurately characterize stripe noise [47]. Therefore, our method can estimate abundance maps from HS images contaminated by mixed noise including dense stripe noise.

B. Optimization Algorithm

To solve Problem (15) by an algorithm based on P-PDS, we need to transform Problem (15) into Problem (1). First, using the indicator functions,[3] Problem (15) are rewritten as follows:

$$\min_{\mathbf{A},\mathbf{S},\mathbf{L}} \|\mathbf{A}\|_{1,2,r} + \lambda_1\|\mathbf{D}(\mathbf{A})\|_1 + \lambda_2\mathcal{R}(\mathbf{K}\circ\mathbf{E}(\mathbf{A})) + \lambda_3\|\mathbf{L}\|_1 + \iota_{\mathbb{R}_+^{m\times n}}(\mathbf{A}) + \iota_{\mathcal{B}_{F,\varepsilon}^{\mathbf{V}}}(\mathbf{EA}+\mathbf{S}+\mathbf{L}) + \iota_{\mathcal{B}_{1,\eta}}(\mathbf{S}) + \iota_{\{\mathbf{O}\}}(\mathbf{D}_v(\mathbf{L})) \tag{18}$$

where $\mathbf{K}\circ\mathbf{E}$ is the composite operator of $\mathbf{K}$ and $\mathbf{E}$, i.e., $\mathbf{K}\circ\mathbf{E}(\mathbf{A}) = \mathbf{K}(\mathbf{EA})$. Introducing auxiliary variables $\mathbf{Z}_1$, $\mathbf{Z}_2$, $\mathbf{Z}_3$, $\mathbf{Z}_4$, and $\mathbf{Z}_5$, we can transform Problem (18) into the following equivalent problem:

$$\min_{\substack{\mathbf{A},\mathbf{S},\mathbf{L},\\ \mathbf{Z}_1,\ldots,\mathbf{Z}_5}} \iota_{\mathbb{R}_+^{m\times n}}(\mathbf{A}) + \iota_{\mathcal{B}_{1,\eta}}(\mathbf{S}) + \lambda_3\|\mathbf{L}\|_1 + \|\mathbf{Z}_1\|_{1,2,r} + \lambda_1\|\mathbf{Z}_2\|_1 + \lambda_2\mathcal{R}(\mathbf{Z}_3) + \iota_{\mathcal{B}_{F,\varepsilon}^{\mathbf{V}}}(\mathbf{Z}_4) + \iota_{\{\mathbf{O}\}}(\mathbf{Z}_5)$$
$$\text{s.t.} \begin{cases} \mathbf{Z}_1 = \mathbf{A} \\ \mathbf{Z}_2 = \mathbf{D}(\mathbf{A}) \\ \mathbf{Z}_3 = \mathbf{K}\circ\mathbf{E}(\mathbf{A}) \\ \mathbf{Z}_4 = \mathbf{EA}+\mathbf{S}+\mathbf{L} \\ \mathbf{Z}_5 = \mathbf{D}_v(\mathbf{L}). \end{cases} \tag{19}$$

[3]If $C\subset\mathbb{R}^{M\times N}$ satisfies $\lambda\mathbf{X}+(1-\lambda)\mathbf{Y}\in C$ for any $\mathbf{X},\mathbf{Y}\in C$ and $\lambda\in[0,1]$, C is a convex set. For a nonempty closed convex set C, the indicator function $\iota_C:\mathbb{R}^{M\times N}\to(-\infty,\infty]$ is defined by $\iota_C(\mathbf{X}) := 0$, if $\mathbf{X}\in C$; $\iota_C(\mathbf{X}) := \infty$, otherwise.

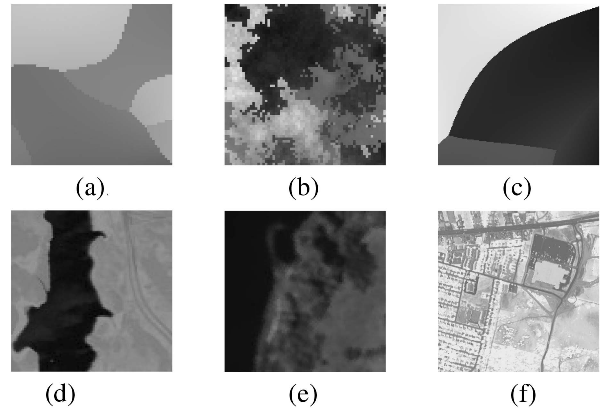

Fig. 3. Original HS images. (a) *Synth 1*. (b) *Synth 2*. (c) *Synth 3*. (d) *Jasper Ridge*. (e) *Samson*. (f) *Urban*.

Finally, by defining $f_1(\mathbf{A}) = \iota_{\mathbb{R}_+^{m\times n}}(\mathbf{A})$, $f_2(\mathbf{S}) = \iota_{\mathcal{B}_{1,\eta}}(\mathbf{S})$, $f_3(\mathbf{L}) = \lambda_3\|\mathbf{L}\|_1$, $g_1(\mathbf{Z}_1) = \|\mathbf{Z}_1\|_{1,2,r}$, $g_2(\mathbf{Z}_2) = \lambda_1\|\mathbf{Z}_2\|_1$, $g_3(\mathbf{Z}_3) = \lambda_2\mathcal{R}(\mathbf{Z}_3)$, $g_4(\mathbf{Z}_4) = \iota_{\mathcal{B}_{F,\varepsilon}^{\mathbf{V}}}(\mathbf{Z}_4)$, and $g_5(\mathbf{Z}_5) = \iota_{\{\mathbf{O}\}}(\mathbf{Z}_5)$, Problem (1) is reduced to Problem (19), i.e., Problem (15). The algorithm for solving Problem (15) is summarized in Algorithm 1. The linear operator $\mathbf{K}$ in steps 4 and 15, and the function $\mathcal{R}$ in step 16 depend on what regularization is adopted, as shown in Table I. The proximity operators are calculated as follows:

$$[\mathrm{prox}_{\gamma\iota_{\mathbb{R}_+^{m\times n}}}(\mathbf{A})]_{i,j} = \max(0, A_{i,j}) \tag{20}$$

$$[\mathrm{prox}_{\gamma\|\cdot\|_1}(\mathbf{A})]_{i,j} = \mathrm{sign}(A_{i,j})\max(|A_{i,j}|-\gamma, 0) \tag{21}$$

$$[\mathrm{prox}_{\gamma\|\cdot\|_{1,2,r}}(\mathbf{A})]_{i,j} = \max\left(1-\frac{\gamma}{\sqrt{\sum_j A_{i,j}^2}}, 0\right)A_{i,j} \tag{22}$$

$$[\mathrm{prox}_{\gamma\|\cdot\|_{1,2,c}}(\mathbf{A})]_{i,j} = \max\left(1-\frac{\gamma}{\sqrt{\sum_i A_{i,j}^2}}, 0\right)A_{i,j} \tag{23}$$

$$\mathrm{prox}_{\gamma\iota_{\mathcal{B}_{F,\varepsilon}^{\mathbf{V}}}}(\mathbf{A}) = \begin{cases} \mathbf{A}, & \text{if } \mathbf{A}\in\mathcal{B}_{F,\varepsilon}^{\mathbf{V}} \\ \mathbf{V}+\frac{\varepsilon(\mathbf{A}-\mathbf{V})}{\|\mathbf{A}-\mathbf{V}\|_F}, & \text{otherwise} \end{cases} \tag{24}$$

$$\mathrm{prox}_{\gamma\iota_{\{\mathbf{O}\}}}(\mathbf{A}) = \mathbf{O}. \tag{25}$$

The proximity operator of $\iota_{\gamma\mathcal{B}_{1,\eta}}$ can be efficiently computed by the ℓ_1-ball projection algorithm [55].

Based on (4), the stepsizes of Algorithm 1 are given as

$$\gamma_1 = \frac{1}{\|\mathbf{I}\|_{\mathrm{op}}^2+\|\mathbf{D}\|_{\mathrm{op}}^2+\|\mathbf{K}\circ\mathbf{E}\|_{\mathrm{op}}^2+\|\mathbf{E}\|_{\mathrm{op}}^2}, \gamma_2 = \frac{1}{\|\mathbf{I}\|_{\mathrm{op}}^2}$$
$$\gamma_3 = \frac{1}{\|\mathbf{I}\|_{\mathrm{op}}^2+\|\mathbf{D}_v\|_{\mathrm{op}}^2}, \gamma_4 = \frac{1}{3}. \tag{26}$$

An identity matrix of any size satisfies $\|\mathbf{I}\|_{\mathrm{op}} = 1$. The operator norm $\|\mathbf{E}\|_{\mathrm{op}}$ is equal to its maximum singular value $\sigma_1(\mathbf{E})$. The operator norms of the other linear operators are not easy to obtain,[4] but they are suppressed by $\|\mathbf{D}_v\|_{\mathrm{op}}\leq 2$, $\|\mathbf{D}_b\|_{\mathrm{op}}\leq 2$, $\|\mathbf{D}\|_{\mathrm{op}}\leq 2\sqrt{2}$, $\|\mathbf{K}\circ\mathbf{E}\|_{\mathrm{op}}\leq\|\mathbf{K}\|_{\mathrm{op}}\|\mathbf{E}\|_{\mathrm{op}}$, $\|\mathbf{D}\circ$

[4]Note that the difference operators are not implemented as matrices. Therefore, we cannot easily obtain the singular values of the matrices representing the difference operators.

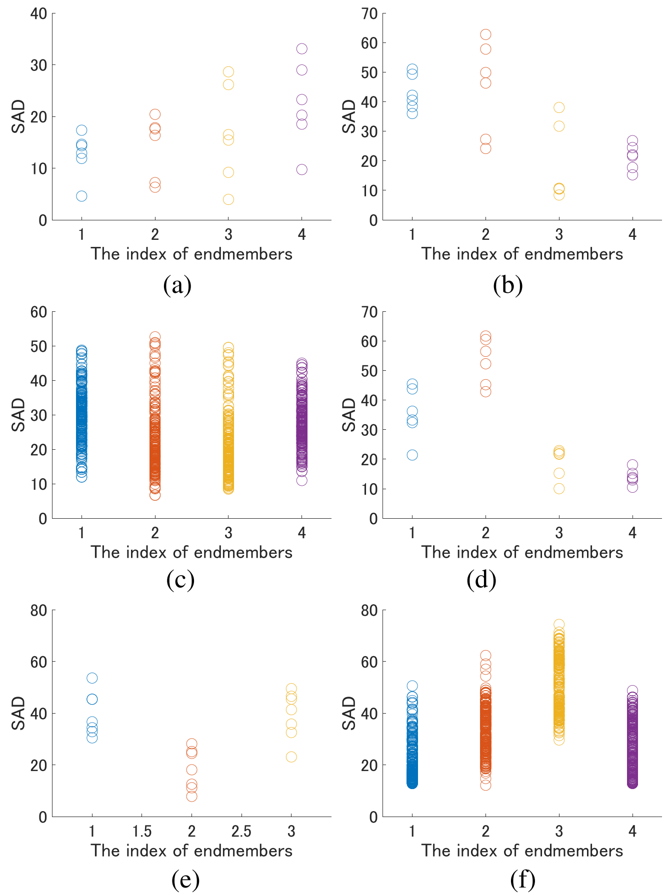

Fig. 4. Distributions of SADs between the spectra of endmembers present in HS images and the spectra of the other endmembers. Vertical axes indicate the SADs. Horizontal axes indicate the indexes of the endmembers present in the HS images. The SAD values tend to be the same for all datasets. (a) *Synth 1*. (b) *Synth 2*. (c) *Synth 3*. (d) *Jasper Ridge*. (e) *Samson*. (f) *Urban*.

$\mathbf{D}_b\|_{\mathrm{op}} \leq \|\mathbf{D}\|_{\mathrm{op}}\|\mathbf{D}_b\|_{\mathrm{op}}$, and $\|\mathbf{C}_\omega\|_{\mathrm{op}} \leq \sqrt{32+8\omega^2}$. By substituting these upper bounds into (5), the specific stepsizes are given, as shown in Table II. This stepsizes design method allows us to avoid the stepsize adjustment for Algorithm 1.

C. Computational Complexity

In general, the computational complexity of our algorithm varies depending on what function and linear operator are used as an image-domain regularization. Our method adopts three image-domain regularizations: 1) HTV, 2) SSTV, and 3) HSSTV. The computational complexities of linear operators and functions including all the image-domain regularizations are given in Table III. From these results, we derive the computational complexities of each step as follows.

1) The complexities of Steps 4, 15, 17, and 18 are $O(nml)$.
2) The complexities of Steps 5, 6, 11, 12, 13, and 14 are $O(nm)$.
3) The complexities of Steps 7, 9, 10, 16, 19, 20, 21, and 22 are $O(nl)$.
4) The complexity of Step 8 is $O(nl \log nl)$.

Therefore, the complexity for each iteration of the algorithm is $O(nl \max\{m, \log nl\})$.

IV. Experiments

We demonstrate the effectiveness of the proposed nonblind unmixing method, i.e., RHUIDR through comprehensive experiments using two synthetic and two real HS images. Specifically, these experiments aim to validate that:

1) RHUIDR achieves good unmixing performance due to image-domain regularizations;
2) RHUIDR is robust to mixed noise, including stripe noise.

As described in the introduction, existing unmixing methods are classified into blind and nonblind, depending on whether the endmember library is given or not. Due to the different assumptions and the fact that blind unmixing methods require a nonblind unmixing step to obtain an initial estimate, it is difficult to fairly compare nonblind unmixing methods with blind ones. Therefore, we mainly compare RHUIDR with nonblind unmixing methods. Specifically, we compare RHUIDR with seven state-of-the-art nonblind unmixing methods and one state-of-the-art blind unmixing method: the collaborative sparse unmixing by variable splitting and augmented Lagrangian (CLSUnSAL) [13], the HS unmixing using joint-sparsity and total variation (JSTV) [19],[5] the row-sparsity spectral unmixing via total variation (RSSUn-TV) [22], the local-global-based sparse regression unmixing (LGSU) [16], the HS unmixing using deep image prior (UnDIP) [28],[6] the endmember-guided unmixing network (EGU-Net) [37],[7] the robust dual spatial weighted sparse unmixing (RDSWSU) [25], and the multidimensional low-rank representation-based sparse HS unmixing (MdLRR) [31].[8] To perform the experiments, we reimplemented the program codes of CLSUnSAL, RSSUn-TV, and RDSWSU. The program code of LGSU was downloaded from a web page, which is no longer accessible as of January 2024. EGU-Net uses the number of endmembers in a target HS image. In our experiments, since the number of endmembers in a target HS image is assumed to be unknown, we set it as the number of endmembers in the endmember libraries of datasets described later. Table IV shows the assumption and the types of noise considered in each method.

A. Datasets Description

We used six datasets for experiments. In all datasets, their endmember libraries were composed of the spectral signatures of the endmembers in ground-truth HS images and other spectral signatures. This is to simulate the real-world situation where we give an endmember library by including more spectral signatures than the components of the target HS image, as assumed in many references of nonblind unmixing.

1) Synthetic HS Image 1 (Synth 1): We generated the first synthetic HS image with a size of $64 \times 64 \times 224$ using the

[5]The code is available at https://jp.mathworks.com/matlabcentral/fileexchange/56831-hyperspectral-unmixing-and-denoising?s_tid=FX_rc1_behav

[6]The code is available at https://github.com/BehnoodRasti/UnDIP

[7]The code is available at https://github.com/danfenghong/IEEE_TNNLS_EGU-Net

[8]The code is available at https://huangjie-uestc.github.io/

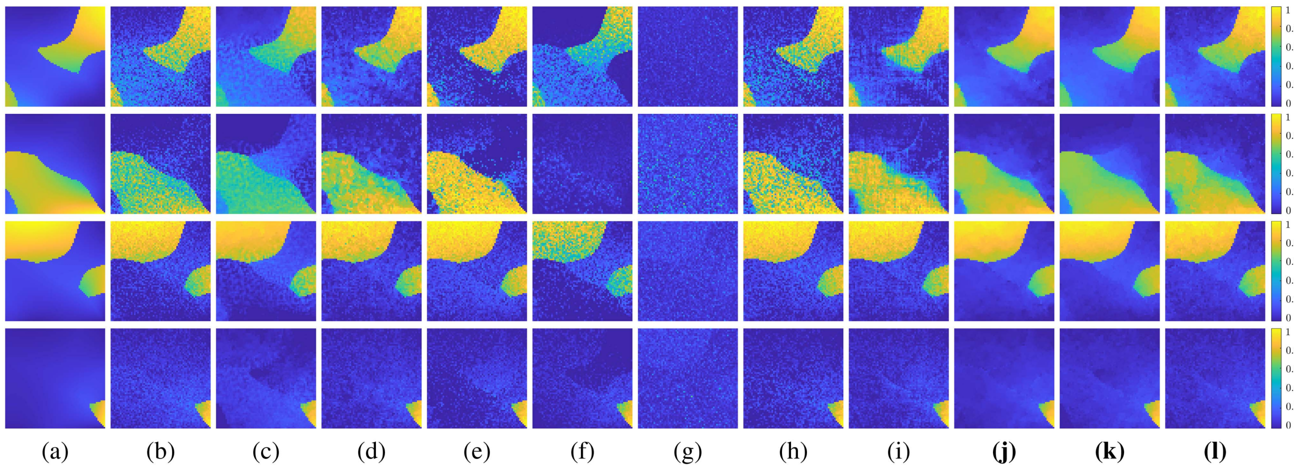

Fig. 5. Unmixing results of abundance maps for the *Synth 1* experiments in Case 2. (a) Original abundance maps. (b) CLSUnSAL [13]. (c) JSTV [19]. (d) RSSUn-TV [22]. (e) LGSU [16]. (f) UnDIP [28]. (g) EGU-Net [37]. (h) RDSWSU [25]. (i) MdLRR [31]. (j) **RHUIDR (HTV)**. (k) **RHUIDR (SSTV)**. (l) **RHUIDR (HSSTV)**.

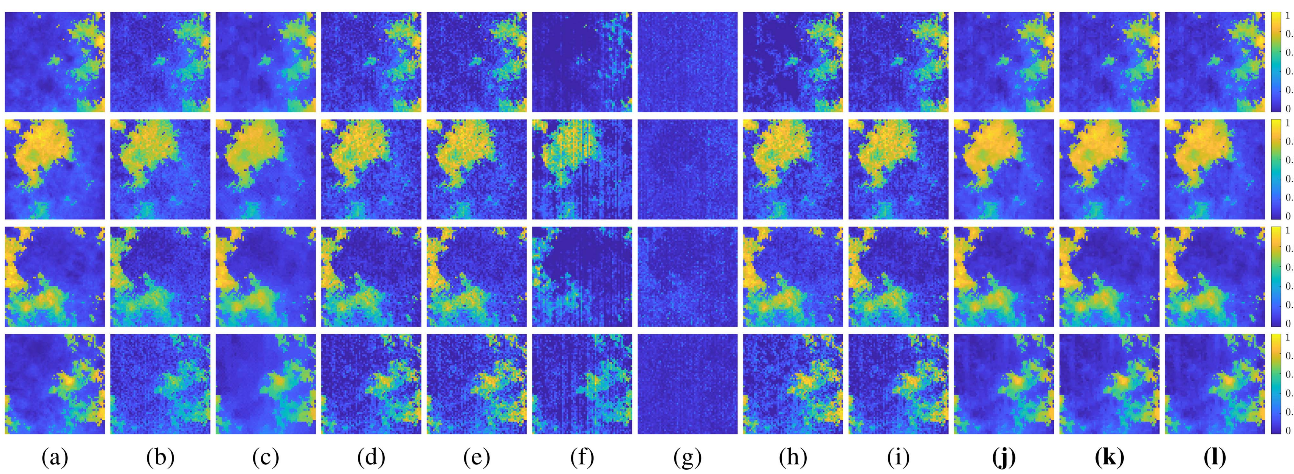

Fig. 6. Unmixing results of abundance maps for the *Synth 2* experiments in Case 5. (a) Original abundance maps. (b) CLSUnSAL [13]. (c) JSTV [19]. (d) RSSUn-TV [22]. (e) LGSU [16]. (f) UnDIP [28]. (g) EGU-Net [37]. (h) RDSWSU [25]. (i) MdLRR [31]. (j) **RHUIDR (HTV)**. (k) **RHUIDR (SSTV)**. (l) **RHUIDR (HSSTV)**.

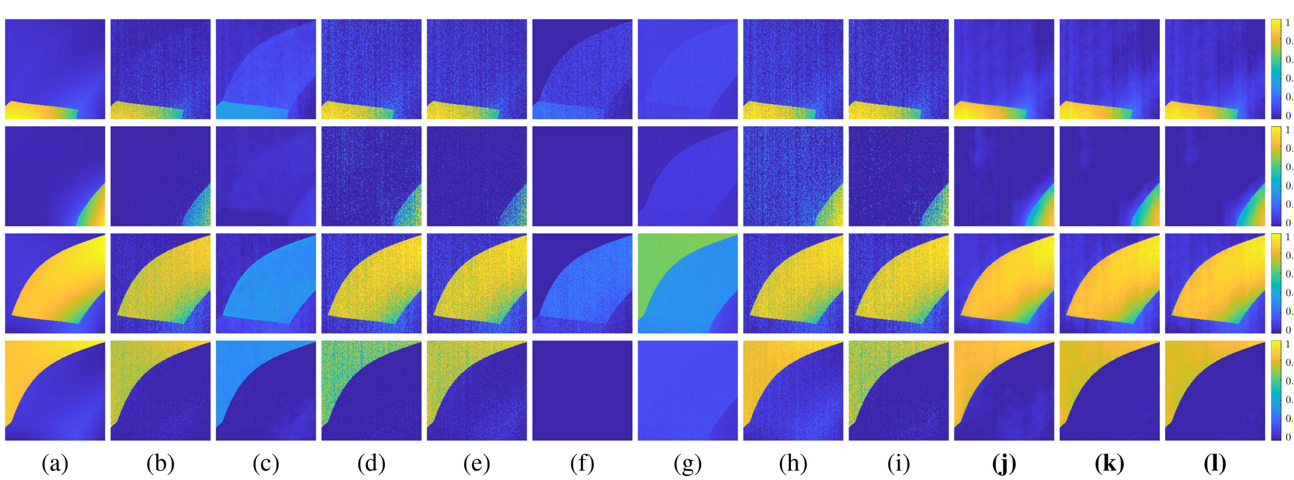

Fig. 7. Unmixing results of abundance maps for the *Synth 3* experiments in Case 8. (a) Original abundance maps. (b) CLSUnSAL [13]. (c) JSTV [19]. (d) RSSUn-TV [22]. (e) LGSU [16]. (f) UnDIP [28]. (g) EGU-Net [37]. (h) RDSWSU [25]. (i) MdLRR [31]. (j) **RHUIDR (HTV)**. (k) **RHUIDR (SSTV)**. (l) **RHUIDR (HSSTV)**.

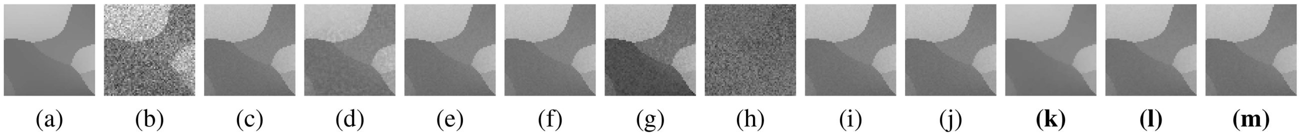

Fig. 8. Reconstructed HS image results for the *Synth 1* experiments in Case 2. (a) Original HS image. (b) Noisy image. (c) CLSUnSAL [13]. (d) JSTV [19]. (e) RSSUn-TV [22]. (f) LGSU [16]. (g) UnDIP [28]. (h) EGU-Net [37]. (i) RDSWSU [25]. (j) MdLRR [31]. (k) **RHUIDR (HTV)**. (l) **RHUIDR (SSTV)**. (m) **RHUIDR (HSSTV)**.

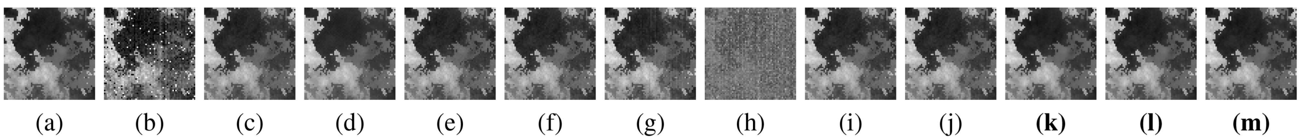

Fig. 9. Reconstructed HS image results for the *Synth 2* experiments in Case 5. (a) Original HS image. (b) Noisy image. (c) CLSUnSAL [13]. (d) JSTV [19]. (e) RSSUn-TV [22]. (f) LGSU [16]. (g) UnDIP [28]. (h) EGU-Net [37]. (i) RDSWSU [25]. (j) MdLRR [31]. (k) **RHUIDR (HTV)**. (l) **RHUIDR (SSTV)**. (m) **RHUIDR (HSSTV)**.

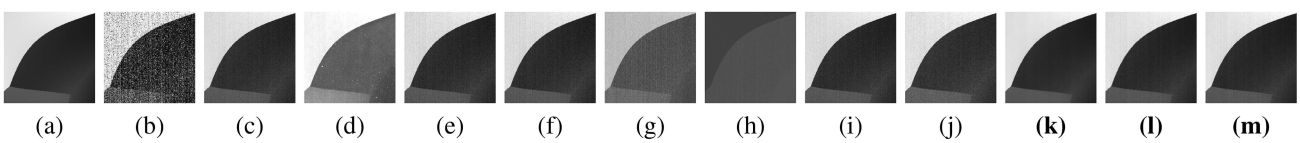

Fig. 10. Reconstructed HS image results for the *Synth 3* experiments in Case 8. (a) Original HS image. (b) Noisy image. (c) CLSUnSAL [13]. (d) JSTV [19]. (e) RSSUn-TV [22]. (f) LGSU [16]. (g) UnDIP [28]. (h) EGU-Net [37]. (i) RDSWSU [25]. (j) MdLRR [31]. (k) **RHUIDR (HTV)**. (l) **RHUIDR (SSTV)**. (m) **RHUIDR (HSSTV)**.

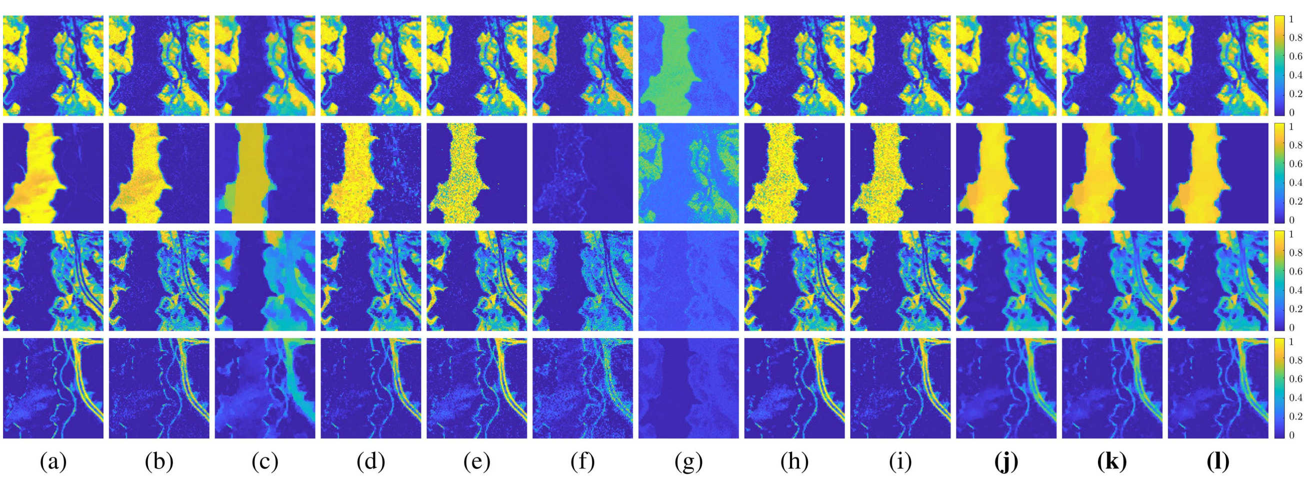

Fig. 11. Unmixing results of abundance maps for the *Jasper Ridge* experiments in Case 2. (a) Original abundance maps. (b) CLSUnSAL [13]. (c) JSTV [19]. (d) RSSUn-TV [22]. (e) LGSU [16]. (f) UnDIP [28]. (g) EGU-Net [37]. (h) RDSWSU [25]. (i) MdLRR [31]. (j) **RHUIDR (HTV)**. (k) **RHUIDR (SSTV)**. (l) **RHUIDR (HSSTV)**.

HYperspectral Data Retrieval and Analysis (HYDRA) toolbox,[9] which is developed by the Computational Intelligence group at the University of the Basque Country. An endmember library consists of ten spectral signatures with 224 bands from the U.S. Geological Survey (USGS) Spectral Library.[10] From the endmember library, we randomly selected four endmembers and generated four original abundance maps with the spatial size of 64×64 using the Legendre method. Fig. 3(a) shows one band of the generated image.

[9]https://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Imagery_Synthesis_tools_for_MATLAB, accessed on Feb. 5, 2023

[10]https://www.usgs.gov/programs/usgs-library, accessed on Aug. 7, 2023

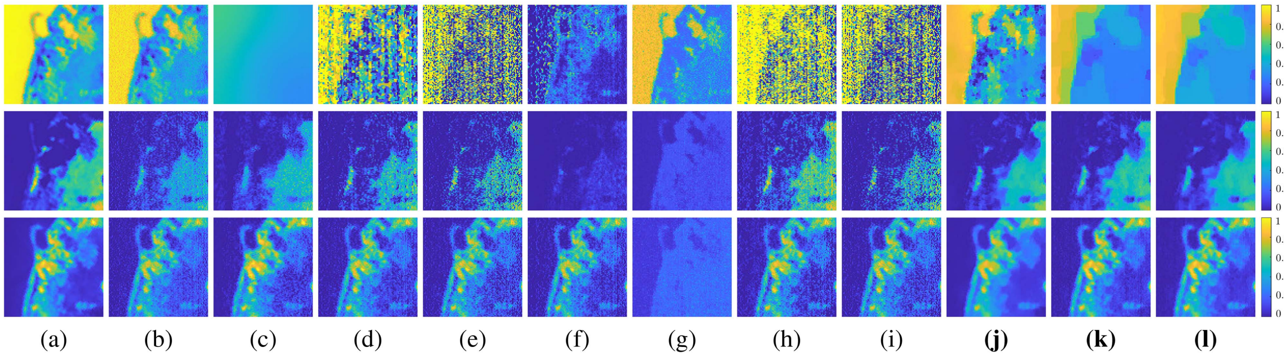

Fig. 12. Unmixing results of abundance maps for the *Samson* experiments in Case 6. (a) Original abundance maps. (b) CLSUnSAL [13]. (c) JSTV [19]. (d) RSSUn-TV [22]. (e) LGSU [16]. (f) UnDIP [28]. (g) EGU-Net [37]. (h) RDSWSU [25]. (i) MdLRR [31]. (j) **RHUIDR (HTV)**. (k) **RHUIDR (SSTV)**. (l) **RHUIDR (HSSTV)**.

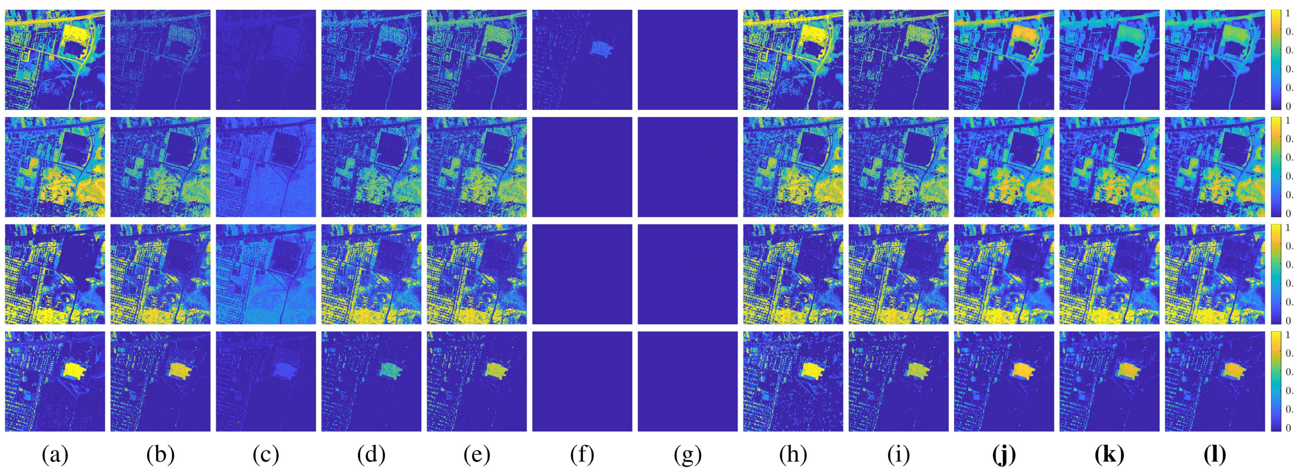

Fig. 13. Unmixing results of abundance maps for the *Urban* experiments in Case 8. (a) Original abundance maps. (b) CLSUnSAL [13]. (c) JSTV [19]. (d) RSSUn-TV [22]. (e) LGSU [16]. (f) UnDIP [28]. (g) EGU-Net [37]. (h) RDSWSU [25]. (i) MdLRR [31]. (j) **RHUIDR (HTV)**. (k) **RHUIDR (SSTV)**. (l) **RHUIDR (HSSTV)**.

2) Synthetic HS Image 2 (Synth 2): We also generated the second synthetic HS image with a size of $64 \times 64 \times 224$ using the HYDRA toolbox. An endmember library consists of ten spectral signatures with 224 bands from the USGS Spectral Library. From the endmember library, we randomly selected four endmembers and generated four original abundance maps with a spatial size of 64×64 using the spherical Gaussian method. Fig. 3(b) shows one band of the generated image.

*3) Synthetic HS Image 3 (*Synth 3*):* We generated the third synthetic HS image with a size of $64 \times 64 \times 224$ using the HYDRA toolbox. An endmember library consists of 240 spectral signatures with 224 bands from the USGS Spectral Library. From the endmember library, we randomly selected four endmembers and generated four original abundance maps with a spatial size of 200×200 using the Legendre method. Fig. 3(c) shows one band of the generated image.

*4) Real HS Image 1 (*Jasper Ridge*): Jasper Ridge* image [see Fig. 3(d)] is captured using an AVIRIS sensor in a rural area in California, USA. The spatial size of the original data is 512×614 pixels, and each pixel holds spectral information in 224 bands ranging from 380–2500 nm. After removing several noisy bands and cropping the image, we obtained the image with 100×100 pixels and 198 bands. *Jasper Ridge* contains four major endmembers: "road," "soil," "water," and "tree" [56]. Adding the six endmembers from the USGS Spectral Library, we used ten endmembers for the experiments.

*5) Real HS Image 2 (*Samson*): Samson* [see Fig. 3(e)] is often used for unmixing. The spatial size of the original data is 952×952 pixels, and each pixel holds spectral information in 156 bands covering the wavelengths from 401 to 889 nm. After cropping the image, we obtained the image with 95×95 pixels. *Samson* contains three major endmembers: "soil," "tree," and "water." Adding the seven endmembers from the USGS Spectral Library, we used ten endmembers for the experiments.

*6) Real HS Image 3 (*Urban*): Urban* [see Fig. 3(f)] was collected by the Hyperspectral Digital Imagery Collection Experiment (HYDICE) over an urban area at Copperas Cove, Texas, USA. The dataset has been widely used in the field of HS unmixing. The latest data version was issued by the Geospatial Research Laboratory and Engineer Research and Development Center in 2015. The image consists of 307×307 pixels with 210 spectral bands in the wavelength from 400 to 2500 nm with a spectral resolution of 10 nm at a ground sampling distance of 2 m. Due to water absorption and atmospheric effects, we

TABLE V
HYPERPARAMETER SETTINGS IN EACH METHOD

Methods	Parameter search range
CLSUnSAL [13]	$\lambda \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}\}$
JSTV [19]	$\lambda_1 \in \{10^{-2}, 10^{-1}, 10^{0}, 10^{1}\}$, $\lambda_2 \in \{10^{-2}, 10^{-1}, 10^{0}, 10^{1}\}$, $\lambda_3 \in \{10^{-1}, 10^{0}, 10^{1}\}$
RSSUn-TV [22]	$\lambda \in \{10^{-2}, 10^{-1}, 10^{0}, 10^{1}\}$, $\lambda_{TV} \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$
LGSU [16]	$\lambda_g \in \{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}\}$, $\lambda_l \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$
UnDIP [28]	–
EGU-Net [37]	–
RDSWSU [25]	$\lambda \in \{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}\}$
MdLRR [31]	$\lambda \in \{10^{-3}, 10^{-2}, 10^{-1}\}$, $\tau \in \{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}\}$
RHUIDR (Ours)	$\lambda_1 \in \{10^{-2}, 10^{-1}, 10^{0}, 10^{1}\}$, $\lambda_2 \in \{10^{-2}, 10^{-1}, 10^{0}, 10^{1}\}$, $\alpha \in \{0.95, 0.98\}$

reduced 210 bands to 162 bands by removing several noisy bands. *Samson* contains four major endmembers: "Asphalt," "Grass," "Tree," and "Roof." Adding the 236 endmembers from the USGS Spectral Library, we used 240 endmembers for the experiments.

Fig. 4 plots the distributions of the spectral angle distance (SAD) values between the spectra of endmembers present in target HS images and the spectra of the other endmembers. The SAD values tend to be the same for all the datasets. In this regard, the difficulty of unmixing is not expected to change.

B. Experimental Setup

HS images are often degraded by mixed noise in real-noise scenarios. Thus, we consider the following eight combinations of i.i.d and non-i.i.d. Gaussian noise with different standard deviations σ, salt-and-pepper noise with different rate $p_{\mathbf{S}}$, and stripe noise in both synthetic and real data experiments.

Case 1 (i.i.d. Gaussian noise): The observed HS image is contaminated by white Gaussian noise with the standard deviation $\sigma = 0.05$.

Case 2 (Higher-level i.i.d. Gaussian noise): The observed HS image is contaminated by white Gaussian noise with the standard deviation $\sigma = 0.1$.

Case 3 (i.i.d. Gaussian noise + salt-and-pepper noise): The observed HS image is contaminated by white Gaussian noise with the standard deviation $\sigma = 0.05$ and salt-and-pepper noise with the rate $p_{\mathbf{S}} = 0.05$.

Case 4 (i.i.d. Gaussian noise + higher-rate salt-and-pepper noise): The observed HS image is contaminated by white Gaussian noise with the standard deviation $\sigma = 0.05$ and salt-and-pepper noise with the rate $p_{\mathbf{S}} = 0.1$.

Case 5 (i.i.d. Gaussian noise + salt-and-pepper noise + stripe noise): The observed HS image is contaminated by white Gaussian noise with the standard deviation $\sigma = 0.05$ and salt-and-pepper noise with the rate $p_{\mathbf{S}} = 0.05$. In addition, the observed HS image is corrupted by vertical stripe noise whose intensity is random in the range $[-0.3, 0.3]$.

Case 6 (i.i.d. higher level Gaussian noise + salt-and-pepper noise + stripe noise): The observed HS image is contaminated by white Gaussian noise with the standard deviation $\sigma = 0.1$ and salt-and-pepper noise with the rate $p_{\mathbf{S}} = 0.05$. In addition, the observed HS image is corrupted by vertical stripe noise whose intensity is random in the range $[-0.3, 0.3]$.

Case 7 (Non-i.i.d. Gaussian noise): The observed HS image is contaminated by non-i.i.d. white Gaussian noise. Specifically, we corrupt each band $\mathbf{h}_i$ by the standard deviation σ_i randomly chosen in the range $[0.1, 0.2]$.

Case 8 (Non-i.i.d. Gaussian noise + salt-and-pepper noise + stripe noise): The observed HS image is contaminated by non-i.i.d. white Gaussian noise and salt-and-pepper noise with the rate $p_{\mathbf{S}} = 0.05$. In addition, the observed HS image is corrupted by vertical stripe noise whose intensity is random in the range $[-0.3, 0.3]$. The non-i.i.d. white Gaussian noise is the same as in Case 7.

The hyperparameters of CLSUnSAL, JSTV, RSSUn-TV, LGSU, RDSWSU, and MdLRR were adjusted to obtain the highest SRE value for each noise case and for each dataset in the ranges shown in Table V. UnDIP and EGU-Net have no parameter to adjust, and we followed the experimental procedures in shown their references. The stopping criteria of the existing methods were determined according to their references. RHUIDR has hyperparameters λ_1, λ_2, λ_3, η, and ε. The hyperparameters λ_1 and λ_2 were adjusted to obtain the highest SRE value in the range shown in Table V. The parameter λ_3 was set to a constant value 1 because it is not very sensitive to performance thanks to the flatness constraint [the fourth constraint of (15)]. The hyperparameter η was set as $\eta = 0.5\alpha_\eta p_{\mathbf{S}} nl$ with $\alpha_\eta = 0.9$. The hyperparameter ε was set as $\varepsilon = \alpha_\sigma \sigma \sqrt{(1-p_{\mathbf{S}})nl}$ for i.i.d. Gaussian noise cases and set as $\varepsilon = \alpha_\sigma \sqrt{(1-p_{\mathbf{S}})nl \sum_i^l \sigma_i}$ for non-i.i.d. Gaussian noise cases with α_σ adjusted in the range shown in Table V. As the parameter of HSSTV, we adopted $\omega = 0.05$, which is recommended in [42]. The maximum iteration and the stopping criterion were set to 50 000 and $\|\mathbf{A}^{(t+1)} - \mathbf{A}^{(t)}\|_F / \|\mathbf{A}^{(t+1)}\|_F \leq 10^{-5}$, respectively.

For the quantitative evaluation of abundance maps, we used the signal reconstruction error (SRE)

$$\mathrm{SRE[dB]} = 10 \log_{10} \left(\frac{\|\bar{\mathbf{A}}\|_F^2}{\|\bar{\mathbf{A}} - \hat{\mathbf{A}}\|_F^2} \right) \tag{27}$$

the root-mean-square error (RMSE)

$$\mathrm{RMSE} = \sqrt{\frac{1}{mn} \|\bar{\mathbf{A}} - \hat{\mathbf{A}}\|_F^2} \tag{28}$$

and the probability of success (Ps)

$$\mathrm{Ps} = P \left(\frac{\|\bar{\mathbf{a}}_i - \hat{\mathbf{a}}_{i,j}\|_2^2}{\|\bar{\mathbf{a}}_i\|_2^2} \leq \text{threshold} \right) \tag{29}$$

where $\bar{\mathbf{A}}$ and $\hat{\mathbf{A}}$ denote the true and estimated abundance maps, respectively. In addition, $\mathbf{a}_i$ is the ith pixel of $\mathbf{A}$ (i.e., $\mathbf{a}_i$ is the ith column vector of $\mathbf{A}$). SRE and RMSE evaluate the difference between the true and estimated abundance maps, with larger SRE or smaller RMSE indicating better-estimated performance. Ps is the probability that the relative error is less than a certain threshold. This threshold is a criterion for how close the true and estimated abundance should be to be considered successful. In setting the threshold, most of the literature, e.g., in [13], [23],

TABLE VI
SRE, RMSE, Ps, MPSNR, AND MSSIM IN THE EXPERIMENTS USING *SYNTH 1*

Noise	Metrics	Methods										
		CLSUnSAL [13]	JSTV [19]	RSSUn-TV [22]	LGSU [16]	UnDIP [28]	EGU-Net [37]	RDSWSU [25]	MdLRR [31]	**RHUIDR (HTV)**	**RHUIDR (SSTV)**	**RHUIDR (HSSTV)**
Case 1	Setup	$\lambda = 10^{-1}$	$\lambda_1 = 10^1$, $\lambda_2 = 10^1$, $\lambda_3 = 10^1$	$\lambda = 10^1$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^{-1}$, $\lambda_l = 10^{-2}$	–	–	$\lambda = 10^{-2}$	$\lambda = 10^{-3}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\varepsilon = 0.95$	$\lambda_1 = 10^1$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$
	SRE	16.22	15.73	22.15	16.73	2.88	-5.52	16.38	18.96	**24.31**	21.84	23.52
	RMSE	0.0385	0.0388	0.0197	0.0391	0.1417	0.2489	0.0398	0.0284	**0.0153**	0.0200	0.0167
	Ps	**1.00**	**1.00**	**1.00**	**1.00**	0.59	0.01	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	45.23	38.35	46.53	45.09	23.92	15.32	45.17	45.59	**54.23**	49.34	47.74
	MSSIM	0.9775	0.9164	0.9840	0.9768	0.8096	0.4255	0.9766	0.9804	**0.9988**	0.9937	0.9886
Case 2	Setup	$\lambda = 10^0$	$\lambda_1 = 10^1$, $\lambda_2 = 10^1$, $\lambda_3 = 10^1$	$\lambda = 10^1$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0$, $\lambda_l = 10^{-1}$	–	–	$\lambda = 10^{-2}$	$\lambda = 10^{-2}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\varepsilon = 0.95$	$\lambda_1 = 10^1$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	12.57	11.35	14.22	14.91	0.42	-5.59	13.12	16.24	**21.64**	20.19	20.84
	RMSE	0.0569	0.0615	0.0485	0.0484	0.1892	0.2510	0.0567	0.0393	**0.0206**	0.0240	0.0226
	Ps	0.98	**1.00**	**1.00**	0.99	0.50	0.01	0.97	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	40.17	34.04	39.90	39.74	19.76	15.25	39.67	39.91	**49.20**	43.39	41.70
	MSSIM	0.9356	0.8270	0.9338	0.9298	0.6906	0.3899	0.9261	0.9327	**0.9959**	0.9735	0.9572
Case 3	Setup	$\lambda = 10^0$	$\lambda_1 = 10^1$, $\lambda_2 = 10^1$, $\lambda_3 = 10^1$	$\lambda = 10^{-2}$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0$, $\lambda_l = 10^{-1}$	–	–	$\lambda = 10^0$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^1$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$
	SRE	11.28	15.18	7.63	13.13	-1.03	-5.62	11.58	14.64	**23.92**	21.06	23.73
	RMSE	0.0646	0.0411	0.1051	0.0583	0.2085	0.2512	0.0790	0.0494	**0.0159**	0.0218	0.0162
	Ps	0.97	**1.00**	0.85	0.98	0.32	0.01	0.96	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	37.77	38.32	37.09	37.30	17.82	15.24	36.15	37.82	**50.54**	48.43	46.82
	MSSIM	0.9066	0.9204	0.8957	0.8919	0.6237	0.3827	0.9239	0.9087	**0.9949**	0.9922	0.9860
Case 4	Setup	$\lambda = 10^0$	$\lambda_1 = 10^1$, $\lambda_2 = 10^1$, $\lambda_3 = 10^1$	$\lambda = 10^{-2}$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0$, $\lambda_l = 10^{-1}$	–	–	$\lambda = 10^0$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^1$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	9.10	14.79	4.45	10.19	-1.62	-5.66	10.25	12.23	**23.59**	21.89	23.31
	RMSE	0.0809	0.0429	0.1558	0.0797	0.2123	0.2526	0.0918	0.0636	**0.0166**	0.0199	0.0172
	Ps	0.93	**1.00**	0.48	0.94	0.31	0.01	0.91	0.98	**1.00**	**1.00**	**1.00**
	MPSNR	34.66	37.77	33.63	33.94	19.02	15.23	33.82	34.27	**47.92**	47.58	45.63
	MSSIM	0.8556	0.9084	0.8286	0.8275	0.6136	0.3774	0.8746	0.8448	0.9893	**0.9900**	0.9810
Case 5	Setup	$\lambda = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\lambda_3 = 10^0$	$\lambda = 10^0$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0$, $\lambda_l = 10^{-1}$	–	–	$\lambda = 10^0$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\varepsilon = 0.95$	$\lambda_1 = 10^1$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^1$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$
	SRE	10.48	13.43	8.03	12.19	-1.11	-5.61	11.28	13.96	**21.38**	21.02	19.96
	RMSE	0.0703	0.0507	0.0964	0.0641	0.2058	0.2522	0.0819	0.0527	**0.0211**	0.0218	0.0244
	Ps	0.96	**1.00**	0.86	0.97	0.31	0.01	0.95	0.99	**1.00**	**1.00**	**1.00**
	MPSNR	36.97	36.20	36.18	36.53	18.10	15.22	35.62	36.98	**50.17**	45.92	45.99
	MSSIM	0.8951	0.9374	0.8800	0.8778	0.6443	0.3822	0.9151	0.8939	**0.9983**	0.9886	0.9893
Case 6	Setup	$\lambda = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\lambda_3 = 10^0$	$\lambda = 10^1$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0$, $\lambda_l = 10^{-1}$	–	–	$\lambda = 10^0$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^1$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$
	SRE	9.72	10.91	6.50	12.21	0.22	-5.65	11.21	13.31	**17.14**	14.86	16.80
	RMSE	0.0764	0.0670	0.1173	0.0656	0.1763	0.2521	0.0812	0.0563	**0.0339**	0.0425	0.0350
	Ps	0.93	**1.00**	0.71	0.97	0.37	0.01	0.95	0.99	**1.00**	**1.00**	**1.00**
	MPSNR	35.82	31.08	35.01	35.58	19.28	15.23	35.23	35.76	41.22	**42.52**	42.03
	MSSIM	0.8661	0.8674	0.8454	0.8515	0.5572	0.3818	0.8849	0.8610	0.9560	**0.9741**	0.9660
Case 7	Setup	$\lambda = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\lambda_3 = 10^0$	$\lambda = 10^{-2}$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^{-1}$, $\lambda_l = 10^{-1}$	–	–	$\lambda = 10^0$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\varepsilon = 0.98$	$\lambda_1 = 10^1$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^1$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$
	SRE	10.49	10.68	9.73	12.92	-0.87	-5.63	12.19	14.60	16.71	16.66	**17.48**
	RMSE	0.0715	0.0700	0.0803	0.0607	0.1991	0.2516	0.0734	0.0493	0.0357	0.0351	**0.0323**
	Ps	0.94	0.99	0.93	0.98	0.34	0.01	0.97	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	36.88	30.34	35.99	36.31	20.45	15.28	35.56	36.73	39.64	39.58	**39.78**
	MSSIM	0.8779	0.8063	0.8578	0.8612	0.5501	0.3838	0.9005	0.8740	0.9339	0.9378	**0**.9406
Case 8	Setup	$\lambda = 10^0$	$\lambda_1 = 10^1$, $\lambda_2 = 10^1$, $\lambda_3 = 10^1$	$\lambda = 10^{-2}$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0$, $\lambda_l = 10^{-1}$	–	–	$\lambda = 10^0$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$
	SRE	8.78	8.81	5.53	11.15	-1.87	-5.64	11.01	12.05	**14.96**	11.89	14.08
	RMSE	0.0851	0.0798	0.1314	0.0732	0.2121	0.2527	0.0839	0.0642	**0.0427**	0.0592	0.0468
	Ps	0.91	0.99	0.65	0.96	0.23	0.01	0.94	0.98	**1.00**	**1.00**	**1.00**
	MPSNR	34.38	31.16	33.42	34.02	19.49	15.25	34.03	34.15	40.36	37.76	**41.30**
	MSSIM	0.8292	0.7416	0.7969	0.8067	0.6387	0.3791	0.8517	0.8148	0.9518	0.9118	**0**.9653

The bold values are best values.

[24], and [25], regards unmixing to be successful when $\|\mathbf{a}_i - \hat{\mathbf{a}}_{i,j}\|_2^2/\|\mathbf{a}_i\|_2^2 < 3.16$ (i.e., $10 * \log_{10}(\|\mathbf{a}_i - \hat{\mathbf{a}}_{i,j}\|_2^2/\|\mathbf{a}_i\|_2^2) < 5$ [dB]). Therefore, in this research, the threshold was also set as 3.16.

For the quantitative evaluation of the reconstructed HS images, we used the mean peak signal-to-noise ratio overall bands (MPSNR)

$$\mathrm{MPSNR[dB]} = \frac{1}{l}\sum_{i=1}^{l} 10\log_{10}\left(\frac{n}{\|\bar{H}_{i,j} - \hat{H}_{i,j}\|_F^2}\right) \tag{30}$$

where $\bar{\mathbf{H}}$ and $\hat{\mathbf{H}}$ are the ground-truth and reconstructed HS images, respectively. In addition, we adopted the mean structural similarity overall bands (MSSIM) [57]

$$\mathrm{MSSIM} = \frac{1}{l}\sum_{i=1}^{l} \mathrm{SSIM}(\bar{\mathbf{H}}_i, \hat{\mathbf{H}}_i) \tag{31}$$

where $\bar{\mathbf{H}}_i$ are $\hat{\mathbf{H}}_i$ are the ith bands of $\bar{\mathbf{H}}$ and $\hat{\mathbf{H}}$, respectively. Higher MPSNR and MSSIM values indicate better reconstruction results.

C. Experimental Results With Synthetic HS Images

Tables VI–VIII show the SRE, RMSE, Ps, MPSNR, and MSSIM results for *Synth 1*, *Synth 2*, and *Synth 3*, respectively. The best and second-best results are highlighted in bold and underlined, respectively. CLSUnSAL, JSTV, RSSUn-TV, and

TABLE VII
SRE, RMSE, Ps, MPSNR, AND MSSIM IN THE EXPERIMENTS USING *SYNTH 2*

Noise	Metrics	Methods										
		CLSUnSAL [13]	JSTV [19]	RSSUn-TV [22]	LGSU [16]	UnDIP [28]	EGU-Net [37]	RDSWSU [25]	MdLRR [31]	**RHUIDR (HTV)**	**RHUIDR (SSTV)**	**RHUIDR (HSSTV)**
Case 1	Setup	$\lambda = 10^1$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\lambda_3 = 10^0$	$\lambda = 10^1$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0$, $\lambda_l = 10^{-3}$	–	–	$\lambda = 10^{-2}$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	18.06	16.65	19.03	19.78	1.88	-5.02	20.80	19.81	**21.11**	20.78	20.99
	RMSE	0.0270	0.0318	0.0252	0.0236	0.1518	0.2272	0.0212	0.0235	**0.0198**	0.0206	0.0201
	Ps	**1.00**	**1.00**	**1.00**	**1.00**	0.55	0.00	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	41.37	32.54	41.80	41.60	27.09	15.79	41.90	41.71	**42.84**	42.39	42.55
	MSSIM	0.9889	0.9435	0.9870	0.9864	0.8702	0.3498	0.9871	0.9876	**0.9911**	0.9891	0.9897
Case 2	Setup	$\lambda = 10^1$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\lambda_3 = 10^0$	$\lambda = 10^{-2}$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0$, $\lambda_l = 10^{-4}$	–	–	$\lambda = 10^{-2}$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	15.49	13.97	13.99	16.01	0.60	-5.08	16.68	15.62	**17.50**	17.23	17.38
	RMSE	0.0358	0.0425	0.0449	0.0365	0.1760	0.2308	0.0342	0.0377	**0.0297**	0.0308	0.0302
	Ps	**1.00**	0.99	**1.00**	**1.00**	0.48	0.00	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	38.11	30.06	37.29	37.46	24.74	15.25	37.58	37.52	**39.00**	38.35	38.51
	MSSIM	0.9735	0.8983	0.9645	0.9654	0.8380	0.2937	0.9667	0.9665	**0.9787**	0.9731	0.9743
Case 3	Setup	$\lambda = 10^1$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\lambda_3 = 10^0$	$\lambda = 10^{-2}$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0$, $\lambda_l = 10^{-2}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	11.62	16.37	10.81	12.36	1.69	-5.08	14.32	12.50	**20.80**	20.46	20.71
	RMSE	0.0529	0.0326	0.0620	0.0529	0.1586	0.2319	0.0436	0.0514	**0.0204**	0.0213	0.0207
	Ps	**1.00**	**1.00**	0.97	0.98	0.62	0.00	**1.00**	0.99	**1.00**	**1.00**	**1.00**
	MPSNR	33.78	32.31	33.71	33.78	27.65	14.98	34.19	33.89	**42.24**	41.90	42.06
	MSSIM	0.9525	0.9444	0.9419	0.9421	0.8395	0.2635	0.9477	0.9445	**0.9902**	0.9883	0.9889
Case 4	Setup	$\lambda = 10^1$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\lambda_3 = 10^0$	$\lambda = 10^{-2}$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0$, $\lambda_l = 10^{-2}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	8.49	15.93	7.29	9.18	-0.08	-5.08	11.89	9.17	**19.02**	18.65	18.92
	RMSE	0.0723	0.0344	0.0910	0.0732	0.1784	0.2332	0.0565	0.0722	**0.0251**	0.0262	0.0254
	Ps	0.97	**1.00**	0.88	0.94	0.43	0.00	0.98	0.95	**1.00**	**1.00**	**1.00**
	MPSNR	30.06	31.42	29.91	30.03	24.84	14.71	30.48	30.07	**41.30**	40.80	40.97
	MSSIM	0.9143	0.9299	0.8993	0.9001	0.7894	0.2472	0.9096	0.9025	**0.9871**	0.9844	0.9852
Case 5	Setup	$\lambda = 10^1$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\lambda_3 = 10^0$	$\lambda = 10^{-2}$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0$, $\lambda_l = 10^{-2}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	11.95	17.36	11.27	12.68	1.63	-5.09	14.25	12.86	**21.52**	21.27	21.42
	RMSE	0.0513	0.0294	0.0590	0.0512	0.1553	0.2323	0.0439	0.0495	**0.0189**	0.0195	0.0191
	Ps	**1.00**	**1.00**	0.98	0.99	0.53	0.00	**1.00**	0.99	**1.00**	**1.00**	**1.00**
	MPSNR	33.88	32.87	33.84	33.90	25.62	14.95	34.24	34.02	**42.83**	42.51	42.65
	MSSIM	0.9530	0.9509	0.9428	0.9429	0.8397	0.2607	0.9484	0.9453	**0.9909**	0.9890	0.9895
Case 6	Setup	$\lambda = 10^1$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\lambda_3 = 10^0$	$\lambda = 10^{-2}$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^{-1}$, $\lambda_l = 10^{-1}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$
	SRE	11.08	13.75	9.03	11.44	-0.74	-5.10	13.62	11.43	**17.96**	17.55	17.81
	RMSE	0.0564	0.0434	0.0776	0.0599	0.1962	0.2331	0.0477	0.0584	**0.0282**	0.0296	0.0288
	Ps	0.99	0.99	0.92	0.97	0.39	0.00	0.99	0.98	**1.00**	**1.00**	**1.00**
	MPSNR	33.06	29.63	32.71	32.84	26.02	14.78	33.28	32.93	**38.55**	38.14	38.24
	MSSIM	0.9401	0.8945	0.9254	0.9282	0.8051	0.2515	0.9327	0.9284	**0.9798**	0.9724	0.9732
Case 7	Setup	$\lambda = 10^1$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\lambda_3 = 10^0$	$\lambda = 10^{-2}$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0$, $\lambda_l = 10^{-2}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	12.85	11.74	10.75	13.58	0.68	-5.10	14.65	13.25	15.49	15.68	**16.05**
	RMSE	0.0477	0.0531	0.0649	0.0483	0.1714	0.2326	0.0437	0.0495	0.0370	0.0362	**0.0348**
	Ps	**1.00**	0.99	0.96	0.99	0.46	0.00	0.99	0.99	**1.00**	**1.00**	**1.00**
	MPSNR	35.24	27.62	34.22	34.53	25.30	14.92	34.93	34.63	**36.76**	35.77	36.11
	MSSIM	0.9488	0.8577	0.9334	0.9365	0.8282	0.2464	0.9425	0.9382	**0.9664**	0.9531	0.9570
Case 8	Setup	$\lambda = 10^1$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\lambda_3 = 10^0$	$\lambda = 10^{-2}$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0$, $\lambda_l = 10^{-1}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$
	SRE	9.67	10.96	7.13	10.72	3.34	-5.09	12.50	10.12	12.99	14.26	**14.60**
	RMSE	0.0655	0.0576	0.0976	0.0650	0.1275	0.2333	0.0547	0.0682	0.0471	0.0415	**0.0405**
	Ps	0.98	0.98	0.85	0.96	0.66	0.00	0.98	0.95	0.98	0.99	0.99
	MPSNR	31.72	26.70	31.17	31.50	27.93	14.73	32.01	31.48	34.17	35.12	**35.21**
	MSSIM	0.9172	0.8258	0.8977	0.9054	0.8409	0.2389	0.9102	0.9029	**0.9575**	0.9517	0.9525

The bold values are best values.

LGSU were not good in all cases. For *Synth 3*, the unmixing performance of JSTV was degraded. This may have been an issue with the algorithm because the value of the optimization problem was not fully reduced.[11] The results of RDSWSU and MdLRR were better than CLSUnSAL, JSTV, RSSUn-TV, and LGSU. However, their performance dropped when HS images were contaminated with sparse noise and stripe noise (Cases 3–6, and 8). UnDIP and EGU-Net yielded worse results than the other existing methods. This is because UnDIP and EGU-Net do not capture the sparsity of abundance maps. In contrast, RHUIDR yielded the best SRE, RMSE, Ps, MPSNR, and MSSIM values in the cases where the HS image is contaminated with noise that can be handled by the existing methods (Cases 1 and 2 for CLSUnSAL, RSSUn-TV, and LGSU, and Cases 1–4 for JSTV), except for Case 1 using *Synth 1*. This indicates that the image-domain regularizations can improve the unmixing performance. In addition, RHUIDR achieved the best performance in the other cases (Cases 5 and 6). This is due to the fact that RHUIDR can handle all three types of noise. In the comparison of the

[11] We used the program code implemented by the authors. Neither varying the step size nor increasing the maximum number of iterations improved the results.

TABLE VIII
SRE, RMSE, PS, MPSNR, AND MSSIM IN THE EXPERIMENTS USING *SYNTH 3*

Noise	Metrics	Methods										
		CLSUnSAL [13]	JSTV [19]	RSSUn-TV [22]	LGSU [16]	UnDIP [28]	EGU-Net [37]	RDSWSU [25]	MdLRR [31]	**RHUIDR (HTV)**	**RHUIDR (SSTV)**	**RHUIDR (HSSTV)**
Case 1	Setup	$\lambda=10^0$	$\lambda_1=10^1$, $\lambda_2=10^{-2}$, $\lambda_3=10^1$	$\lambda=10^0$, $\lambda_{TV}=10^{-2}$	$\lambda_g=10^0$, $\lambda_l=10^{-2}$	–	–	$\lambda=10^{-1}$	$\lambda=10^{-1}$, $\tau=10^0$	$\lambda_1=10^{-1}$, $\lambda_2=10^0$, $\varepsilon=0.95$	$\lambda_1=10^0$, $\lambda_2=10^{-1}$, $\varepsilon=0.95$	$\lambda_1=10^{-1}$, $\lambda_2=10^{-1}$, $\varepsilon=0.95$
	SRE	21.68	-0.00	23.03	22.90	-0.49	-3.64	25.54	24.21	**30.26**	26.38	27.58
	RMSE	0.0045	1.8879	0.0039	0.0040	0.0380	0.0520	0.0030	0.0034	**0.0017**	0.0026	0.0023
	Ps	**1.00**	0.00	**1.00**	**1.00**	0.52	0.02	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	42.76	-44.42	42.72	43.14	18.91	12.43	45.54	43.10	**56.00**	46.05	46.49
	MSSIM	0.9614	0.7841	0.9556	0.9587	0.6812	0.6928	0.9721	0.9647	**0.9997**	0.9822	0.9834
Case 2	Setup	$\lambda=10^0$	$\lambda_1=10^{-2}$, $\lambda_2=10^{-1}$, $\lambda_3=10^1$	$\lambda=10^1$, $\lambda_{TV}=10^{-2}$	$\lambda_g=10^0$, $\lambda_l=10^{-2}$	–	–	$\lambda=10^{-1}$	$\lambda=10^{-1}$, $\tau=10^0$	$\lambda_1=10^{-1}$, $\lambda_2=10^0$, $\varepsilon=0.95$	$\lambda_1=10^0$, $\lambda_2=10^{-1}$, $\varepsilon=0.95$	$\lambda_1=10^{-1}$, $\lambda_2=10^{-1}$, $\varepsilon=0.98$
	SRE	16.96	-0.00	15.10	16.82	-3.02	-0.35	20.97	18.83	**28.67**	22.34	22.72
	RMSE	0.0076	1.6608	0.0095	0.0080	0.0410	0.0346	0.0050	0.0063	**0.0020**	0.0041	0.0040
	Ps	**1.00**	0.00	**1.00**	**1.00**	0.02	0.53	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	37.22	-43.74	36.07	36.67	18.26	15.57	40.13	35.15	**52.34**	40.45	42.56
	MSSIM	0.8721	0.4809	0.8288	0.8415	0.7765	0.8055	0.9052	0.8377	**0.9991**	0.9361	0.9624
Case 3	Setup	$\lambda=10^0$	$\lambda_1=10^{-2}$, $\lambda_2=10^{-2}$, $\lambda_3=10^1$	$\lambda=10^1$, $\lambda_{TV}=10^{-2}$	$\lambda_g=10^1$, $\lambda_l=10^{-1}$	–	–	$\lambda=10^{-1}$	$\lambda=10^{-1}$, $\tau=10^0$	$\lambda_1=10^{-1}$, $\lambda_2=10^0$, $\varepsilon=0.95$	$\lambda_1=10^0$, $\lambda_2=10^{-1}$, $\varepsilon=0.95$	$\lambda_1=10^{-1}$, $\lambda_2=10^{-1}$, $\varepsilon=0.98$
	SRE	13.12	-0.00	10.65	13.66	-3.83	-4.33	15.63	13.97	**29.66**	25.91	26.93
	RMSE	0.0111	2.1084	0.0152	0.0112	0.0429	0.0479	0.0090	0.0105	**0.0018**	0.0028	0.0025
	Ps	0.99	0.00	0.98	0.97	0.02	0.02	**1.00**	0.99	**1.00**	**1.00**	**1.00**
	MPSNR	32.00	-45.39	31.33	32.49	20.44	12.87	34.41	29.86	**55.27**	45.23	47.14
	MSSIM	0.7730	0.5410	0.7136	0.7836	0.5486	0.6956	0.8264	0.6874	**0.9996**	0.9783	0.9870
Case 4	Setup	$\lambda=10^0$	$\lambda_1=10^{-2}$, $\lambda_2=10^{-2}$, $\lambda_3=10^1$	$\lambda=10^1$, $\lambda_{TV}=10^{-2}$	$\lambda_g=10^1$, $\lambda_l=10^{-1}$	–	–	$\lambda=10^1$	$\lambda=10^{-1}$, $\tau=10^0$	$\lambda_1=10^{-1}$, $\lambda_2=10^0$, $\varepsilon=0.95$	$\lambda_1=10^0$, $\lambda_2=10^{-1}$, $\varepsilon=0.95$	$\lambda_1=10^{-1}$, $\lambda_2=10^{-1}$, $\varepsilon=0.98$
	SRE	9.59	-0.00	6.88	9.73	-6.48	-3.74	10.14	9.44	**30.92**	25.52	26.68
	RMSE	0.0156	2.0454	0.0225	0.0161	0.0489	0.0432	0.0167	0.0169	**0.0016**	0.0029	0.0025
	Ps	0.96	0.00	0.87	0.94	0.00	0.04	0.92	0.95	**1.00**	**1.00**	**1.00**
	MPSNR	27.91	-45.13	27.34	28.16	16.13	15.71	30.11	25.25	**55.21**	44.34	45.62
	MSSIM	0.6530	0.4900	0.5783	0.6521	0.4271	0.7979	0.8082	0.5151	**0.9996**	0.9730	0.9803
Case 5	Setup	$\lambda=10^0$	$\lambda_1=10^{-1}$, $\lambda_2=10^0$, $\lambda_3=10^1$	$\lambda=10^1$, $\lambda_{TV}=10^{-2}$	$\lambda_g=10^1$, $\lambda_l=10^{-1}$	–	–	$\lambda=10^{-1}$	$\lambda=10^{-1}$, $\tau=10^0$	$\lambda_1=10^0$, $\lambda_2=10^1$, $\varepsilon=0.95$	$\lambda_1=10^0$, $\lambda_2=10^{-1}$, $\varepsilon=0.98$	$\lambda_1=10^0$, $\lambda_2=10^{-1}$, $\varepsilon=0.98$
	SRE	12.80	-0.00	10.15	13.66	-5.29	-4.01	15.24	13.47	**24.96**	21.46	21.54
	RMSE	0.0115	1.7139	0.0161	0.0111	0.0461	0.0470	0.0094	0.0112	**0.0031**	0.0046	0.0045
	Ps	0.99	0.00	0.97	0.97	0.00	0.04	**1.00**	0.99	**1.00**	**1.00**	**1.00**
	MPSNR	31.76	-44.03	31.01	32.21	15.25	13.15	34.15	29.38	**47.52**	41.18	41.51
	MSSIM	0.7597	0.4738	0.6970	0.7688	0.4620	0.7016	0.8153	0.6696	**0.9980**	0.9507	0.9550
Case 6	Setup	$\lambda=10^0$	$\lambda_1=10^{-2}$, $\lambda_2=10^{-1}$, $\lambda_3=10^1$	$\lambda=10^1$, $\lambda_{TV}=10^{-2}$	$\lambda_g=10^1$, $\lambda_l=10^{-1}$	–	–	$\lambda=10^{-1}$	$\lambda=10^{-1}$, $\tau=10^0$	$\lambda_1=10^0$, $\lambda_2=10^0$, $\varepsilon=0.98$	$\lambda_1=10^0$, $\lambda_2=10^{-1}$, $\varepsilon=0.98$	$\lambda_1=10^0$, $\lambda_2=10^{-1}$, $\varepsilon=0.98$
	SRE	11.84	-0.00	8.75	12.21	-6.97	-4.64	14.26	11.85	**24.39**	20.38	20.36
	RMSE	0.0127	1.5402	0.0188	0.0126	0.0530	0.0520	0.0105	0.0134	**0.0033**	0.0051	0.0051
	Ps	0.98	0.00	0.93	0.96	0.00	0.01	**1.00**	0.98	**1.00**	**1.00**	**1.00**
	MPSNR	30.96	-43.10	30.01	31.31	17.37	12.17	33.31	27.65	**47.02**	38.23	38.86
	MSSIM	0.7132	0.2564	0.6349	0.7118	0.4976	0.6436	0.7763	0.5911	**0.9956**	0.9029	0.9176
Case 7	Setup	$\lambda=10^0$	$\lambda_1=10^{-1}$, $\lambda_2=10^0$, $\lambda_3=10^1$	$\lambda=10^1$, $\lambda_{TV}=10^{-2}$	$\lambda_g=10^1$, $\lambda_l=10^{-2}$	–	–	$\lambda=10^{-1}$	$\lambda=10^{-1}$, $\tau=10^0$	$\lambda_1=10^{-1}$, $\lambda_2=10^0$, $\varepsilon=0.95$	$\lambda_1=10^0$, $\lambda_2=10^{-1}$, $\varepsilon=0.95$	$\lambda_1=10^0$, $\lambda_2=10^{-1}$, $\varepsilon=0.95$
	SRE	14.14	-0.00	10.63	13.86	-4.27	-4.21	17.18	14.05	**25.57**	20.01	20.00
	RMSE	0.0103	1.4475	0.0158	0.0113	0.0464	0.0527	0.0077	0.0109	**0.0029**	0.0054	0.0054
	Ps	0.99	0.00	0.97	0.97	0.01	0.01	**1.00**	0.99	**1.00**	**1.00**	**1.00**
	MPSNR	33.96	-42.56	32.28	33.57	19.63	12.39	36.47	29.24	**48.64**	37.09	38.06
	MSSIM	0.7684	0.2893	0.6830	0.7356	0.4921	0.6608	0.8178	0.6416	**0.9969**	0.8716	0.8980
Case 8	Setup	$\lambda=10^0$	$\lambda_1=10^1$, $\lambda_2=10^{-2}$, $\lambda_3=10^1$	$\lambda=10^1$, $\lambda_{TV}=10^{-2}$	$\lambda_g=10^1$, $\lambda_l=10^{-3}$	–	–	$\lambda=10^{-1}$	$\lambda=10^{-1}$, $\tau=10^0$	$\lambda_1=10^0$, $\lambda_2=10^0$, $\varepsilon=0.98$	$\lambda_1=10^0$, $\lambda_2=10^{-1}$, $\varepsilon=0.98$	$\lambda_1=10^0$, $\lambda_2=10^{-1}$, $\varepsilon=0.98$
	SRE	10.67	-0.00	6.95	9.74	-6.72	-3.78	12.74	9.49	**21.12**	18.67	18.69
	RMSE	0.0144	1.0417	0.0234	0.0167	0.0517	0.0474	0.0126	0.0176	**0.0047**	0.0062	0.0062
	Ps	0.96	0.01	0.84	0.94	0.00	0.04	0.99	0.94	**1.00**	**1.00**	**1.00**
	MPSNR	29.84	-39.41	28.58	29.37	14.55	13.37	32.13	24.96	**43.66**	35.56	36.44
	MSSIM	0.6476	0.4875	0.5494	0.5894	0.4098	0.7328	0.7146	0.4711	**0.9925**	0.8375	0.8696

The bold values are best values.

image-domain regularizations, HTV performed better in almost all cases, and HSSTV performed better in the *Synth 1* and *Synth 2* experiments when HS images were contaminated with non-i.i.d. Gaussian noise (Cases 7 and 8).

Figs. 5–7 show the estimated abundance maps for *Synth 1* in Case 4, for *Synth 2* in Case 5, and for *Synth 3* in Case 8. All the abundance maps of CLSUnSAL, RSSUn-TV, LGSU, and RDSWSU include residual noise in Cases 2, 5, and 8 [Figs. 5, 6, and 7(b), (d), (e), and (h)]. JSTV remained the noise [Fig. 5(c)] or obtained the oversmooth abundance maps [the second abundance map for *Synth 2* in Fig. 6(c)], due to the difficulty of adjusting the parameters balancing the sparsity and piecewise-smoothness of abundance maps. For *Synth 3* [Fig. 7(c)], JSTV obtained the significantly lower abundance maps. This may be due to an algorithm issue. The abundance maps of MdLRR are relatively exact in Case 2 [Fig. 5(i)], but are affected by sparse and stripe noise in Cases 5 and 8 [Figs. 6 and 7(i)]. UnDIP and EGU-Net erroneously estimated that the abundances were high for the endmembers that are not present in the HS images due to the insufficient ability

TABLE IX
SRE, RMSE, Ps, MPSNR, AND MSSIM IN THE EXPERIMENTS USING *JASPER RIDGE*

Noise	Metrics	Methods										
		CLSUnSAL [13]	JSTV [19]	RSSUn-TV [22]	LGSU [16]	UnDIP [28]	EGU-Net [37]	RDSWSU [25]	MdLRR [31]	**RHUIDR (HTV)**	**RHUIDR (SSTV)**	**RHUIDR (HSSTV)**
Case 1	Setup	$\lambda = 10^0$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \lambda_3 = 10^1$	$\lambda = 10^1, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^{-1}, \lambda_l = 10^{-2}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-1}, \tau = 10^0$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \varepsilon = 0.95$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \varepsilon = 0.95$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \varepsilon = 0.95$
	SRE	<u>20.24</u>	15.99	20.12	19.45	3.59	-1.42	20.13	**20.48**	19.07	19.33	19.06
	RMSE	<u>0.0256</u>	0.0412	0.0263	0.0286	0.1518	0.2002	0.0270	**0.0252**	0.0294	0.0286	0.0295
	Ps	**1.00**	**1.00**	**1.00**	**1.00**	0.64	0.07	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	45.22	35.09	45.50	45.60	32.81	18.56	**45.78**	<u>45.65</u>	45.45	45.47	45.45
	MSSIM	0.9877	0.9234	0.9878	0.9871	0.9330	0.4523	0.9877	0.9876	**0.9882**	0.9881	**0.9882**
Case 2	Setup	$\lambda = 10^0$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \lambda_3 = 10^0$	$\lambda = 10^1, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0, \lambda_l = 10^{-2}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-1}, \tau = 10^0$	$\lambda_1 = 10^0, \lambda_2 = 10^0, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \varepsilon = 0.98$
	SRE	14.81	10.14	13.91	13.93	-0.13	-3.23	15.35	14.32	<u>15.49</u>	**15.53**	15.37
	RMSE	0.0468	0.0743	0.0535	0.0527	0.2247	0.2664	0.0470	0.0510	<u>0.0436</u>	**0.0432**	0.0441
	Ps	**0.99**	0.96	0.98	0.98	0.58	0.00	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**
	MPSNR	39.53	32.25	39.45	39.77	27.85	17.65	40.15	39.46	**40.97**	40.31	<u>40.36</u>
	MSSIM	0.9582	0.8809	0.9533	0.9541	0.8343	0.4552	0.9569	0.9512	**0.9732**	0.9634	<u>0.9641</u>
Case 3	Setup	$\lambda = 10^0$	$\lambda_1 = 10^0, \lambda_2 = 10^1, \lambda_3 = 10^1$	$\lambda = 10^1, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0, \lambda_l = 10^{-2}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-1}, \tau = 10^0$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \varepsilon = 0.95$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \varepsilon = 0.95$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \varepsilon = 0.95$
	SRE	10.14	15.82	9.06	8.78	-1.88	-1.89	10.25	8.87	<u>18.76</u>	**18.98**	18.75
	RMSE	0.0751	0.0419	0.0961	0.0948	0.2327	0.1952	0.0872	0.0969	<u>0.0304</u>	**0.0296**	<u>0.0304</u>
	Ps	0.94	**1.00**	0.88	0.89	0.35	0.33	0.92	0.88	**1.00**	**1.00**	**1.00**
	MPSNR	32.74	34.89	32.66	32.74	24.90	19.56	33.50	32.54	**44.47**	**44.47**	**44.47**
	MSSIM	0.8733	0.9195	0.8628	0.8584	0.7162	0.5420	0.8761	0.8561	<u>0.9849</u>	0.9848	**0.9850**
Case 4	Setup	$\lambda = 10^1$	$\lambda_1 = 10^0, \lambda_2 = 10^{-1}, \lambda_3 = 10^0$	$\lambda = 10^1, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0, \lambda_l = 10^{-2}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-1}, \tau = 10^0$	$\lambda_1 = 10^0, \lambda_2 = 10^0, \varepsilon = 0.95$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \varepsilon = 0.95$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \varepsilon = 0.95$
	SRE	6.70	13.03	5.70	5.75	-2.60	-2.05	6.39	5.61	**18.31**	<u>18.29</u>	18.26
	RMSE	0.1005	0.0559	0.1490	0.1377	0.2527	0.1926	0.1420	0.1471	**0.0319**	<u>0.0321</u>	<u>0.0321</u>
	Ps	0.85	0.98	0.66	0.72	0.21	0.33	0.74	0.70	**1.00**	**1.00**	**1.00**
	MPSNR	28.01	35.05	27.98	28.03	23.94	19.71	28.39	27.93	**43.89**	<u>43.39</u>	<u>43.39</u>
	MSSIM	0.7871	0.9343	0.7589	0.7560	0.7056	0.5584	0.7704	0.7532	**0.9836**	<u>0.9803</u>	0.9802
Case 5	Setup	$\lambda = 10^0$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \lambda_3 = 10^0$	$\lambda = 10^1, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0, \lambda_l = 10^{-2}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-1}, \tau = 10^0$	$\lambda_1 = 10^0, \lambda_2 = 10^0, \varepsilon = 0.95$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \varepsilon = 0.95$
	SRE	9.56	10.65	8.27	8.12	0.13	-0.50	9.41	8.21	**16.36**	16.16	<u>16.19</u>
	RMSE	0.0799	0.0716	0.1054	0.1020	0.1976	0.1761	0.0968	0.1043	**0.0394**	0.0401	<u>0.0400</u>
	Ps	0.92	0.96	0.84	0.86	0.50	0.37	0.89	0.85	<u>0.99</u>	<u>0.99</u>	**1.00**
	MPSNR	32.39	32.40	32.26	32.34	27.32	20.03	33.10	32.15	**42.26**	<u>41.63</u>	41.49
	MSSIM	0.8645	0.9018	0.8529	0.8486	0.7606	0.6114	0.8666	0.8464	**0.9791**	<u>0.9744</u>	0.9735
Case 6	Setup	$\lambda = 10^1$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \lambda_3 = 10^0$	$\lambda = 10^1, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0, \lambda_l = 10^{-2}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-1}, \tau = 10^0$	$\lambda_1 = 10^0, \lambda_2 = 10^0, \varepsilon = 0.95$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \varepsilon = 0.95$	$\lambda_1 = 10^0, \lambda_2 = 10^{-1}, \varepsilon = 0.95$
	SRE	9.16	9.13	7.26	7.87	1.51	-1.03	8.60	7.28	**14.39**	14.03	<u>14.19</u>
	RMSE	0.0807	0.0818	0.1204	0.1070	0.1629	0.1836	0.1075	0.1180	**0.0489**	0.0507	<u>0.0500</u>
	Ps	0.93	0.95	0.78	0.84	0.61	0.34	0.85	0.80	**0.99**	**0.99**	**0.99**
	MPSNR	31.64	30.93	31.58	31.83	26.89	20.02	32.42	31.47	**38.82**	37.78	<u>38.04</u>
	MSSIM	0.8543	0.8573	0.8291	0.8287	0.7650	0.5836	0.8441	0.8224	**0.9563**	0.9396	<u>0.9435</u>
Case 7	Setup	$\lambda = 10^1$	$\lambda_1 = 10^0, \lambda_2 = 10^{-1}, \lambda_3 = 10^1$	$\lambda = 10^1, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0, \lambda_l = 10^{-2}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-1}, \tau = 10^0$	$\lambda_1 = 10^0, \lambda_2 = 10^0, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^{-1}, \varepsilon = 0.98$
	SRE	11.51	8.85	9.75	10.24	-0.93	-1.06	11.75	10.19	**13.67**	13.27	<u>13.35</u>
	RMSE	0.0653	0.0911	0.0870	0.0805	0.2345	0.2025	0.0718	0.0822	**0.0531**	0.0551	<u>0.0547</u>
	Ps	**0.98**	0.91	0.90	0.92	0.47	0.35	0.94	0.91	**0.98**	**0.98**	**0.98**
	MPSNR	35.46	26.18	35.69	35.83	25.69	18.70	36.49	35.68	**38.22**	36.94	<u>37.25</u>
	MSSIM	0.9131	0.6888	0.9011	0.9000	0.7413	0.3965	0.9110	0.8988	**0.9502**	0.9250	<u>0.9310</u>
Case 8	Setup	$\lambda = 10^1$	$\lambda_1 = 10^0, \lambda_2 = 10^{-1}, \lambda_3 = 10^1$	$\lambda = 10^{-2}, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^0, \lambda_l = 10^{-2}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-1}, \tau = 10^0$	$\lambda_1 = 10^0, \lambda_2 = 10^0, \varepsilon = 0.95$	$\lambda_1 = 10^0, \lambda_2 = 10^{-2}, \varepsilon = 0.95$	$\lambda_1 = 10^0, \lambda_2 = 10^{-1}, \varepsilon = 0.95$
	SRE	8.39	7.73	6.20	6.75	-0.65	-3.94	7.34	6.14	**12.46**	12.12	<u>12.21</u>
	RMSE	0.0880	0.1014	0.1398	0.1257	0.2172	0.2594	0.1302	0.1387	**0.0604**	0.0623	<u>0.0619</u>
	Ps	0.89	0.87	0.71	0.77	0.40	0.00	0.77	0.73	**0.97**	0.97	0.97
	MPSNR	31.05	25.28	30.86	31.08	25.74	15.92	31.74	30.74	**36.93**	35.67	<u>35.97</u>
	MSSIM	0.8282	0.6531	0.7953	0.7949	0.7198	0.2987	0.8119	0.7891	**0.9371**	0.9055	<u>0.9124</u>

The bold values are best values.

to capture the sparsity of abundance maps, resulting in the generation of inappropriate abundance maps [Figs. 5, 6, and 7(f) and (g)]. In particular, since all the existing methods do not account for stripe noise, they produced the abundance maps that are strongly affected by stripe noise [see Figs. 6 and 7(b), (d)–(g)] or the smoother results than true abundance [Figs. 6 and 7(c)]. In contrast, RHUIDR accurately estimated abundance maps regardless of what type of noise contaminates HS images [Figs. 5, 6, and 7(j)–(l)].

Figs. 8–10 display the reconstructed HS images for *Synth 1* in Case 2, for *Synth 2* in Case 5, and for *Synth 3* in Case 8, respectively. All the existing methods resulted in Gaussian noise remaining in the reconstructed HS images in Case 2 [Fig. 8(c)–(j)]. Moreover, in the reconstructed HS images in Cases 5 and 8 [Figs. 9 and 10(c)–(j)], we can see that residual stripe noise remains. On the other hand, RHUIDR produced clean reconstructed HS images due to the image-domain regularizations.

D. Experiments With Real HS Images

Tables IX–XI show the SRE, RMSE, Ps, MPSNR, and MSSIM results for *Jasper Ridge*, *Samson*, for *Urban*, respectively. The best and second-best results are highlighted in bold

TABLE X
SRE, RMSE, Ps, MPSNR, AND MSSIM IN THE EXPERIMENTS USING *SAMSON*

Noise	Metrics	Methods										
		CLSUnSAL [13]	JSTV [19]	RSSUn-TV [22]	LGSU [16]	UnDIP [28]	EGU-Net [37]	RDSWSU [25]	MdLRR [31]	**RHUIDR (HTV)**	**RHUIDR (SSTV)**	**RHUIDR (HSSTV)**
Case 1	Setup	$\lambda = 10^1$	$\lambda_1 = 10^1, \lambda_2 = 10^0, \lambda_3 = 10^1$	$\lambda = 10^1, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^{-2}, \lambda_l = 10^{-4}$	–	–	$\lambda = 10^{-2}$	$\lambda = 10^{-3}, \tau = 10^0$	$\lambda_1 = 10^{-1}, \lambda_2 = 10^1, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^0, \varepsilon = 0.95$	$\lambda_1 = 10^{-1}, \lambda_2 = 10^0, \varepsilon = 0.98$
	SRE	16.82	0.01	17.52	10.89	2.52	7.27	13.37	13.96	**19.51**	16.48	18.03
	RMSE	0.0338	4.0921	0.0309	0.0678	0.1491	0.0940	0.0539	0.0474	**0.0254**	0.0345	0.0296
	Ps	**1.00**	0.97	**1.00**	0.94	0.56	0.80	0.97	0.99	**1.00**	**1.00**	**1.00**
	MPSNR	34.42	-11.99	45.31	43.54	28.92	24.85	44.90	44.01	43.75	45.52	46.74
	MSSIM	0.9218	0.9600	0.9839	0.9731	0.8409	0.7107	0.9772	0.9774	0.9836	0.9855	0.9907
Case 2	Setup	$\lambda = 10^1$	$\lambda_1 = 10^1, \lambda_2 = 10^0, \lambda_3 = 10^1$	$\lambda = 10^1, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^{-1}, \lambda_l = 10^{-4}$	–	–	$\lambda = 10^{-2}$	$\lambda = 10^{-2}, \tau = 10^0$	$\lambda_1 = 10^{-1}, \lambda_2 = 10^1, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^0, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^1, \varepsilon = 0.98$
	SRE	**18.36**	6.79	11.64	6.46	-1.91	6.42	8.60	7.12	14.51	11.09	12.86
	RMSE	**0.0291**	0.0902	0.0576	0.1120	0.2408	0.0976	0.0960	0.1032	0.0435	0.0590	0.0504
	Ps	**1.00**	0.97	0.99	0.74	0.31	0.77	0.83	0.78	0.99	0.99	0.99
	MPSNR	38.12	29.09	39.23	38.01	23.33	25.39	39.20	37.96	**40.84**	40.45	40.44
	MSSIM	0.9402	0.6970	0.9387	0.9147	0.6756	0.6802	0.9255	0.9166	**0.9719**	0.9561	0.9725
Case 3	Setup	$\lambda = 10^1$	$\lambda_1 = 10^1, \lambda_2 = 10^{-1}, \lambda_3 = 10^1$	$\lambda = 10^1, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^{-1}, \lambda_l = 10^{-3}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-2}, \tau = 10^0$	$\lambda_1 = 10^{-1}, \lambda_2 = 10^1, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^0, \varepsilon = 0.95$	$\lambda_1 = 10^{-1}, \lambda_2 = 10^0, \varepsilon = 0.98$
	SRE	14.35	10.96	8.78	4.48	-2.66	4.13	6.49	4.31	**18.16**	15.61	16.74
	RMSE	0.0446	0.0624	0.0913	0.1695	0.2501	0.1159	0.1660	0.1715	**0.0294**	0.0378	0.0341
	Ps	**1.00**	0.99	0.88	0.53	0.14	0.58	0.62	0.51	**1.00**	**1.00**	**1.00**
	MPSNR	31.57	32.84	31.68	31.21	23.22	21.63	32.82	30.89	43.09	44.34	**45.45**
	MSSIM	0.8178	0.8224	0.8019	0.7713	0.6394	0.5828	0.8190	0.7644	0.9812	0.9807	**0.9868**
Case 4	Setup	$\lambda = 10^1$	$\lambda_1 = 10^1, \lambda_2 = 10^1, \lambda_3 = 10^1$	$\lambda = 10^1, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^{-1}, \lambda_l = 10^{-3}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-2}, \tau = 10^0$	$\lambda_1 = 10^{-1}, \lambda_2 = 10^1, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^0, \varepsilon = 0.95$	$\lambda_1 = 10^0, \lambda_2 = 10^0, \varepsilon = 0.95$
	SRE	10.51	0.02	5.16	3.10	-1.86	-1.41	3.97	2.90	**16.09**	14.57	15.24
	RMSE	0.0653	3.0269	0.1542	0.2343	0.2358	0.1998	0.2263	0.2362	**0.0364**	0.0417	0.0390
	Ps	0.96	0.95	0.60	0.33	0.19	0.18	0.35	0.32	**1.00**	**1.00**	**1.00**
	MPSNR	27.06	-8.81	27.00	26.74	24.75	16.26	27.47	26.59	42.80	43.41	**44.25**
	MSSIM	0.6876	0.9504	0.6646	0.6377	0.6187	0.2036	0.6772	0.6351	0.9805	0.9761	**0.9815**
Case 5	Setup	$\lambda = 10^1$	$\lambda_1 = 10^1, \lambda_2 = 10^{-2}, \lambda_3 = 10^1$	$\lambda = 10^1, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^{-1}, \lambda_l = 10^{-3}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-2}, \tau = 10^0$	$\lambda_1 = 10^0, \lambda_2 = 10^1, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^{-1}, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^0, \varepsilon = 0.98$
	SRE	**13.60**	7.85	7.63	4.19	-0.17	3.94	6.34	4.01	12.91	10.04	12.12
	RMSE	**0.0478**	0.0816	0.1031	0.1754	0.1974	0.1176	0.1667	0.1770	0.0489	0.0648	0.0532
	Ps	0.99	0.98	0.82	0.50	0.36	0.59	0.61	0.49	**1.00**	0.99	**1.00**
	MPSNR	31.31	30.73	31.35	30.90	29.99	22.42	32.49	30.60	**42.74**	40.95	42.48
	MSSIM	0.8107	0.7485	0.7920	0.7622	0.7643	0.6111	0.8128	0.7561	**0.9811**	0.9625	0.9759
Case 6	Setup	$\lambda = 10^1$	$\lambda_1 = 10^1, \lambda_2 = 10^1, \lambda_3 = 10^1$	$\lambda = 10^1, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^{-1}, \lambda_l = 10^{-3}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-2}, \tau = 10^0$	$\lambda_1 = 10^{-1}, \lambda_2 = 10^1, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^{-1}, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^0, \varepsilon = 0.98$
	SRE	**13.59**	5.51	6.56	3.78	-2.24	4.30	5.63	3.67	12.49	9.33	11.97
	RMSE	**0.0484**	0.0993	0.1219	0.1929	0.2439	0.1156	0.1898	0.1932	0.0525	0.0689	0.0543
	Ps	0.99	0.98	0.74	0.46	0.26	0.59	0.56	0.44	0.99	0.98	**1.00**
	MPSNR	30.84	30.86	30.77	30.33	25.92	22.21	31.85	30.07	38.41	37.76	**39.70**
	MSSIM	0.7877	0.7506	0.7611	0.7326	0.6786	0.5919	0.7804	0.7277	0.9559	0.9222	0.9573
Case 7	Setup	$\lambda = 10^1$	$\lambda_1 = 10^1, \lambda_2 = 10^{-2}, \lambda_3 = 10^1$	$\lambda = 10^1, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^{-1}, \lambda_l = 10^{-3}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-2}, \tau = 10^0$	$\lambda_1 = 10^0, \lambda_2 = 10^1, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^1, \varepsilon = 0.98$	$\lambda_1 = 10^0, \lambda_2 = 10^1, \varepsilon = 0.98$
	SRE	**16.71**	4.71	7.68	4.59	-2.67	-0.10	6.43	4.83	13.61	8.38	11.28
	RMSE	**0.0350**	0.1076	0.0892	0.1420	0.2622	0.1946	0.1197	0.1382	0.0466	0.0759	0.0588
	Ps	**1.00**	0.93	0.87	0.61	0.14	0.23	0.67	0.62	**1.00**	0.98	0.99
	MPSNR	35.59	27.13	35.75	34.90	26.96	17.60	36.39	34.79	**39.74**	37.62	38.83
	MSSIM	0.8968	0.6114	0.8757	0.8482	0.7173	0.2423	0.8831	0.8475	**0.9651**	0.9317	0.9612
Case 8	Setup	$\lambda = 10^1$	$\lambda_1 = 10^1, \lambda_2 = 10^1, \lambda_3 = 10^1$	$\lambda = 10^1, \lambda_{TV} = 10^{-2}$	$\lambda_g = 10^{-1}, \lambda_l = 10^{-3}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-2}, \tau = 10^0$	$\lambda_1 = 10^0, \lambda_2 = 10^1, \varepsilon = 0.95$	$\lambda_1 = 10^0, \lambda_2 = 10^{-1}, \varepsilon = 0.95$	$\lambda_1 = 10^{-1}, \lambda_2 = 10^0, \varepsilon = 0.98$
	SRE	**11.52**	2.72	5.16	3.30	-2.20	-1.28	5.23	3.23	10.39	6.26	9.40
	RMSE	**0.0589**	0.1208	0.1400	0.2047	0.2283	0.1958	0.1979	0.2037	0.0630	0.0922	0.0714
	Ps	0.97	0.76	0.63	0.40	0.12	0.23	0.53	0.39	**0.99**	0.84	0.96
	MPSNR	30.31	29.70	30.13	29.70	23.63	16.86	31.25	29.52	37.22	35.18	**37.76**
	MSSIM	0.7595	0.7024	0.7230	0.6970	0.5958	0.2474	0.7493	0.6943	**0.9447**	0.8707	0.9405

The bold values are best values.

and underlined, respectively. In Case 1 of the *Jasper Ridge* experiments, the unmixing performances of CLSUnSAL, RSSUn-TV, LGSU, RDSWSU, MdLRR, and RHUIDR were almost equal. For *Urban*, the unmixing performance of JSTV was degraded similar to the results of *Synth 3*. This may also have been an issue with the algorithm. However, CLSUnSAL, JSTV, RSSUn-TV, LGSU, RDSWSU, and MdLRR performed worse in the other cases than RHUIDR. Since UnDIP and EGU-Net cannot capture the sparsity of abundance maps, their unmixing performance was low regardless of whether HS images are synthetic or real, the size of HS images, and the size of endmember libraries. CLSUnSAL achieved the best SRE and RMSE values for *Samson* in almost all the cases. RDSWSU achieved the best results for *Urban* in almost all the cases. This is because the appropriate weight values for abundance maps were computed due to the proper segmentation. In contrast, RHUIDR achieved the best and second best SRE, RMSE, Ps, MPSNR, and MSSIM values in almost all the cases where the HS image is contaminated by noise that cannot be handled by the existing methods (Cases 3–6, and 8 for CLSUnSAL, RSSUn-TV, LGSU, RDSWSU, and MdLRR, and Cases 5, 6, and 8 for JSTV). In the comparison of the image-domain regularizations, HTV performed better in almost all cases, and HSSTV performed better in the *Samson* experiments when

TABLE XI
SRE, RMSE, Ps, MPSNR, AND MSSIM IN THE EXPERIMENTS USING *URBAN*

Noise	Metrics	Methods										
		CLSUnSAL [13]	JSTV [19]	RSSUn-TV [22]	LGSU [16]	UnDIP [28]	EGU-Net [37]	RDSWSU [25]	MdLRR [31]	**RHUIDR (HTV)**	**RHUIDR (SSTV)**	**RHUIDR (HSSTV)**
Case 1	Setup	$\lambda = 10^0$	$\lambda_1 = 10^{-2}$, $\lambda_2 = 10^{-2}$, $\lambda_3 = 10^1$	$\lambda = 10^0$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^1$, $\lambda_l = 10^{-3}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-2}$, $\tau = 10^0$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$
	SRE	13.07	-0.00	15.62	20.91	0.42	-2.81	**24.94**	18.28	15.92	14.49	14.74
	RMSE	0.0105	2.1963	0.0081	0.0047	0.0336	0.0409	**0.0029**	0.0062	0.0078	0.0090	0.0088
	Ps	0.99	0.11	**1.00**	**1.00**	0.53	0.07	**1.00**	**1.00**	0.99	0.99	0.99
	MPSNR	40.67	-45.31	40.88	42.44	24.71	15.70	**43.78**	40.05	37.93	41.74	41.91
	MSSIM	0.9813	0.6266	0.9806	0.9859	0.8480	0.2748	**0.9894**	0.9791	0.9741	0.9854	0.9863
Case 2	Setup	$\lambda = 10^0$	$\lambda_1 = 10^{-2}$, $\lambda_2 = 10^{-2}$, $\lambda_3 = 10^1$	$\lambda = 10^1$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^1$, $\lambda_l = 10^{-4}$	–	–	$\lambda = 10^{-1}$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$
	SRE	7.33	-0.00	8.91	12.88	-2.49	-7.34	**19.15**	12.98	12.62	10.90	11.37
	RMSE	0.0191	2.1447	0.0165	0.0111	0.0389	0.0569	**0.0057**	0.0113	0.0110	0.0131	0.0126
	Ps	0.81	0.12	0.92	0.98	0.29	0.00	**1.00**	0.96	0.98	0.98	0.98
	MPSNR	34.87	-45.10	34.39	35.69	21.98	13.21	**37.83**	33.83	35.25	35.77	36.05
	MSSIM	0.9360	0.5726	0.9232	0.9382	0.7388	0.3409	**0.9607**	0.9306	0.9571	0.9448	0.9483
Case 3	Setup	$\lambda = 10^0$	$\lambda_1 = 10^1$, $\lambda_2 = 10^{-2}$, $\lambda_3 = 10^1$	$\lambda = 10^1$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^1$, $\lambda_l = 10^{-3}$	–	–	$\lambda = 10^0$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$
	SRE	5.06	-0.00	5.46	10.01	-5.38	-7.92	**15.64**	9.47	15.29	13.79	14.11
	RMSE	0.0233	1.6567	0.0235	0.0149	0.0463	0.0561	0.0085	0.0161	**0.0083**	0.0097	0.0094
	Ps	0.72	0.23	0.74	0.93	0.02	0.00	**0.99**	0.90	**0.99**	**0.99**	**0.99**
	MPSNR	31.03	-42.73	30.72	31.73	20.09	13.05	33.91	29.55	37.38	40.78	**40.98**
	MSSIM	0.8842	0.8106	0.8656	0.8876	0.7273	0.3206	0.9320	0.8551	0.9712	0.9819	**0.9831**
Case 4	Setup	$\lambda = 10^0$	$\lambda_1 = 10^1$, $\lambda_2 = 10^{-2}$, $\lambda_3 = 10^{-1}$	$\lambda = 10^1$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^1$, $\lambda_l = 10^{-2}$	–	–	$\lambda = 10^0$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$
	SRE	2.56	0.00	2.08	6.05	-5.08	-8.40	12.42	5.33	**14.95**	13.76	14.05
	RMSE	0.0290	8.8367	0.0339	0.0219	0.0437	0.0555	0.0119	0.0243	**0.0086**	0.0097	0.0095
	Ps	0.60	0.72	0.43	0.79	0.11	0.00	0.97	0.72	**0.99**	**0.99**	**0.99**
	MPSNR	27.58	-57.72	27.20	28.08	19.23	13.01	29.85	25.67	37.34	39.87	**40.12**
	MSSIM	0.8119	0.5146	0.7824	0.8139	0.6811	0.2982	0.8767	0.7507	0.9713	0.9776	**0.9792**
Case 5	Setup	$\lambda = 10^0$	$\lambda_1 = 10^1$, $\lambda_2 = 10^{-2}$, $\lambda_3 = 10^1$	$\lambda = 10^1$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^1$, $\lambda_l = 10^{-3}$	–	–	$\lambda = 10^0$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^1$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$
	SRE	4.47	0.00	4.44	8.99	-5.94	-8.13	**15.21**	8.34	10.38	9.59	9.59
	RMSE	0.0247	9.6721	0.0262	0.0170	0.0466	0.0558	**0.0089**	0.0181	0.0138	0.0149	0.0148
	Ps	0.69	0.62	0.65	0.89	0.01	0.00	**0.99**	0.86	0.96	0.94	0.94
	MPSNR	30.57	-58.27	30.16	31.14	17.69	13.03	33.48	28.88	30.36	37.42	**37.43**
	MSSIM	0.8743	0.7935	0.8522	0.8736	0.6354	0.3098	0.9244	0.8400	0.8906	0.9677	**0.9679**
Case 6	Setup	$\lambda = 10^0$	$\lambda_1 = 10^1$, $\lambda_2 = 10^{-2}$, $\lambda_3 = 10^1$	$\lambda = 10^1$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^1$, $\lambda_l = 10^{-3}$	–	–	$\lambda = 10^0$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	3.65	0.00	3.20	7.48	-5.80	-8.10	**14.37**	6.85	10.02	8.59	8.69
	RMSE	0.0269	9.5123	0.0303	0.0199	0.0484	0.0559	**0.0098**	0.0212	0.0144	0.0166	0.0165
	Ps	0.64	0.47	0.53	0.83	0.06	0.00	**0.98**	0.79	0.95	0.93	0.93
	MPSNR	29.63	-58.13	29.08	30.01	18.89	13.26	32.55	27.29	**34.34**	33.87	34.18
	MSSIM	0.8490	0.7094	0.8194	0.8423	0.6723	0.3455	0.9066	0.7968	**0.9496**	0.9227	0.9294
Case 7	Setup	$\lambda = 10^0$	$\lambda_1 = 10^{-2}$, $\lambda_2 = 10^0$, $\lambda_3 = 10^1$	$\lambda = 10^1$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^1$, $\lambda_l = 10^{-3}$	–	–	$\lambda = 10^0$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$
	SRE	4.61	-0.00	4.46	9.06	-6.68	-7.88	**16.19**	8.43	10.55	7.97	8.32
	RMSE	0.0253	2.0093	0.0270	0.0171	0.0582	0.0561	**0.0082**	0.0184	0.0136	0.0177	0.0171
	Ps	0.67	0.09	0.61	0.88	0.00	0.00	**0.99**	0.85	0.96	0.91	0.92
	MPSNR	31.60	-44.48	30.64	31.88	12.53	12.99	**34.67**	28.67	32.38	32.70	33.19
	MSSIM	0.8806	0.4875	0.8485	0.8723	0.1260	0.3170	**0.9295**	0.8358	0.9250	0.8999	0.9117
Case 8	Setup	$\lambda = 10^0$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-1}$, $\lambda_3 = 10^1$	$\lambda = 10^1$, $\lambda_{TV} = 10^{-2}$	$\lambda_g = 10^1$, $\lambda_l = 10^{-2}$	–	–	$\lambda = 10^0$	$\lambda = 10^{-1}$, $\tau = 10^0$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$
	SRE	2.55	-0.00	1.55	5.20	-6.84	-8.13	**13.14**	4.47	8.92	6.64	7.32
	RMSE	0.0302	2.1167	0.0380	0.0246	0.0578	0.0559	**0.0113**	0.0274	0.0160	0.0199	0.0186
	Ps	0.58	0.06	0.38	0.71	0.00	0.00	**0.97**	0.65	0.92	0.85	0.88
	MPSNR	28.26	-44.97	27.45	28.50	13.55	13.19	31.20	25.12	**31.82**	31.34	31.62
	MSSIM	0.8057	0.3782	0.7635	0.7943	0.1835	0.3380	0.8736	0.7267	**0.9192**	0.8727	0.8796

The bold values are best values.

HS images were contaminated with non-i.i.d. Gaussian noise (Cases 7 and 8).

Figs. 11–13 show the estimated abundance maps for *Jasper Ridge* in Case 2, for *Samson* in Case 6, and for *Urban* in Case 8, respectively. Although CLSUnSAL, LGSU, RDSWSU, and MdLRR achieved good SRE, RMSE, and MPSNR in Case 2 of the real data experiments, they yielded the abundance maps with residual noise [see Fig. 11(b), (d), (e), (h), and (i)]. JSTV obtained the oversmooth abundance maps [Figs. 11 and 12(c)]. In Cases 6 and 8, the abundance maps estimated by all the existing methods except JSTV include noise, especially stripe noise [see Figs. 12 and 13, (b) (d)–(g)]. This is because they do not handle the stripe noise. In contrast, RHUIDR exactly estimated the abundance maps even under the conditions assumed by the existing methods, e.g., when the observed HS images are only contaminated by Gaussian noise (Fig. 11). Furthermore, RHUIDR estimated the abundance maps by removing not only Gaussian and sparse noise but also stripe noise cleanly.

Figs. 14– 16 display the reconstructed HS images for *Jasper Ridge* in Case 2, for *Samson* in Case 6, and for *Urban* in Case 8. All the existing methods resulted in noise remaining in the reconstructed HS images [Figs. 14, 15, and 16(c)–(j)]. In particular, they cannot handle stripe noise, and thus, did not completely remove it in Case 6 [Figs. 15 and 16(c)–(j)]. On

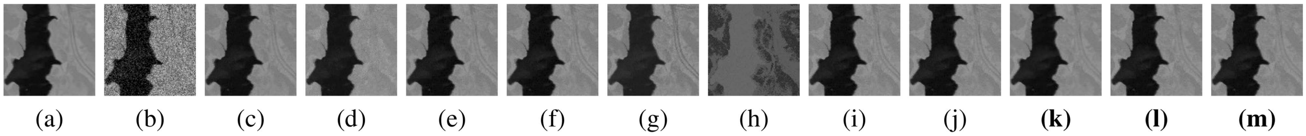

Fig. 14. Reconstructed HS image results for the *Jasper Ridge* experiments in Case 2. (a) Original HS image. (b) Noisy image. (c) CLSUnSAL [13]. (d) JSTV [19]. (e) RSSUn-TV [22]. (f) LGSU [16]. (g) UnDIP [28]. (h) EGU-Net [37]. (i) RDSWSU [25]. (j) MdLRR [31]. (k) **RHUIDR (HTV)**. (l) **RHUIDR (SSTV)**. (m) **RHUIDR (HSSTV)**.

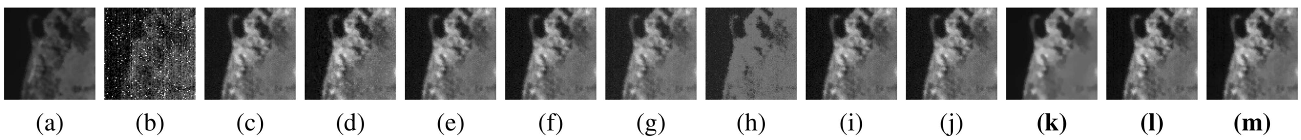

Fig. 15. Reconstructed HS image results for the *Samson* experiments in Case 6. (a) Original HS image. (b) Noisy image. (c) CLSUnSAL [13]. (d) JSTV [19]. (e) RSSUn-TV [22]. (f) LGSU [16]. (g) UnDIP [28]. (h) EGU-Net [37]. (i) RDSWSU [25]. (j) MdLRR [31]. (k) **RHUIDR (HTV)**. (l) **RHUIDR (SSTV)**. (m) **RHUIDR (HSSTV)**.

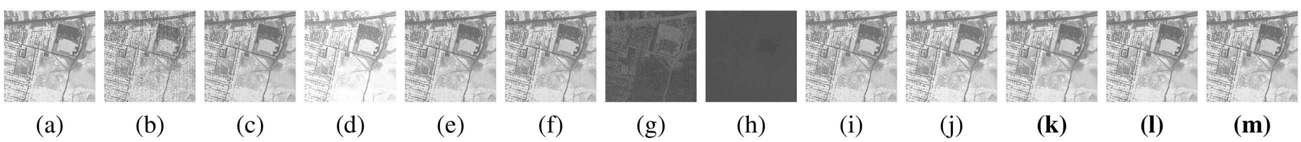

Fig. 16. Reconstructed HS image results for the *Urban* experiments in Case 8. (a) Original HS image. (b) Noisy image. (c) CLSUnSAL [13]. (d) JSTV [19]. (e) RSSUn-TV [22]. (f) LGSU [16]. (g) UnDIP [28]. (h) EGU-Net [37]. (i) RDSWSU [25]. (j) MdLRR [31]. (k) **RHUIDR (HTV)**. (l) **RHUIDR (SSTV)**. (m) **RHUIDR (HSSTV)**.

TABLE XII
AVERAGES OF RUNNING TIMES [S] IN ALL NOISE CASES FOR EACH DATASET

Methods	*Jasper Ridge*	*Samson*	*Urban*
CLSUnSAL [13]	1.359×10^0	9.739×10^{-1}	9.192×10^1
JSTV [19]	7.776×10^1	7.787×10^1	5.709×10^2
RSSUn-TV [22]	1.870×10^0	1.615×10^0	1.111×10^2
LGSU [16]	2.165×10^1	1.720×10^1	6.655×10^2
UnDIP [28]	1.436×10^2	1.030×10^2	2.349×10^3
EGU-Net [37]	1.669×10^1	1.420×10^1	2.360×10^2
RDSWSU [25]	1.239×10^1	9.790×10^0	4.633×10^2
MdLRR [31]	7.268×10^0	6.592×10^0	4.776×10^2
RHUIDR(HTV)	2.252×10^1	4.674×10^1	5.768×10^2
RHUIDR(SSTV)	2.822×10^1	5.372×10^1	8.269×10^2
RHUIDR(HSSTV)	2.440×10^1	5.141×10^1	8.265×10^2

the other hand, RHUIDR reconstructed the HS image cleanly [Figs. 14 and 15(k)–(m)]. This verifies the effectiveness of the image-domain regularization.

E. Comparison of Computational Cost

We measured the actual running times on a Windows 11 computer with an Intel Core i9-13 900 1.0 GHz processor, 32 GB of RAM, and NVIDIA GeForce RTX 4090. In addition, we used MATLAB (R2023b), Python 3.8, and Python 3.7 for CLSUnSAL, JSTV, RSSUn-TV, RDSWSU, MdLRR, and our method, for UnDIP, and for EGU-Net, respectively. The stopping criteria of the comparison methods were set to the values recommended in the papers.

Table XII shows the averages of the running times in all the noise cases for *Jasper Ridge*, *Samson*, and *Urban*. The running time of RHUIDR varies depending on which regularization was employed. When using HTV, SSTV, and HSSTV, RHUIDR took 30 s, 50 s, and 13 min for *Jasper Ridge*, *Samson*, and *Urban*, respectively. Compared with the existing methods, RHUIDR is faster than UnDIP, is the same as LGSU and JSTV, and is slower than CLSUnSAL, RSSUn-TV, EGU-Net, RDSWSU, and MdLRR.

F. Convergence Analysis

In addition, we experimentally analyzed the convergence of our method. Fig. 17 plots the relative error of abundance maps: $\|\mathbf{A}^{(t+1)} - \mathbf{A}^{(t)}\|_{F/}\|\mathbf{A}^{(t)}\|_F$, the objective function values, the Frobenius distance between $\mathbf{V}$ and $\mathbf{EA}^{(t)} + \mathbf{S}^{(t)} + \mathbf{L}^{(t)}$, the ℓ_1 norm of $\mathbf{S}^{(t)}$, and the mean absolute values (MAV) of $\mathbf{D}_v(\mathbf{L}^{(t)})$ for *Jasper Ridge* and *Samson*. The relative error of abundance maps decreased [Fig. 17(a)]. While getting larger as the number of iterations increases, the objective function value asymptotically approaches a certain value [Fig. 17(b)]. This is often found when solving optimization problems involving hard constraints, such as a data fidelity constraint $\mathbf{EA} + \mathbf{S} + \mathbf{L} \in \mathcal{B}^{\mathbf{V}}_{F,\varepsilon}$, a sparsity constraint $\mathbf{S} \in \mathcal{B}_{1,\eta}$, and a flatness constraint $\mathbf{D}_v(\mathbf{L}) = \mathbf{O}$. The ℓ_2 distance and the MAV become smaller, where we can see that

TABLE XIII
SRE, RMSE, Ps, MPSNR, AND MSSIM OF THE ABLATION EXPERIMENTS USING SYNTHETIC DATASETS

Image	Metrics	*Synth 1*				*Synth 2*				*Synth 3*			
		RHUIDR –	**RHUIDR (HTV)**	**RHUIDR (SSTV)**	**RHUIDR (HSSTV)**	RHUIDR –	**RHUIDR (HTV)**	**RHUIDR (SSTV)**	**RHUIDR (HSSTV)**	RHUIDR –	**RHUIDR (HTV)**	**RHUIDR (SSTV)**	**RHUIDR (HSSTV)**
Case 1	Setup	$\lambda_1 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$
	SRE	<u>23.54</u>	**24.31**	21.84	23.52	20.85	**21.11**	20.78	<u>20.99</u>	27.08	**30.26**	26.38	<u>27.58</u>
	RMSE	<u>0.0166</u>	**0.0153**	0.0200	0.0167	0.0204	**0.0198**	0.0206	<u>0.0201</u>	0.0024	**0.0017**	0.0026	<u>0.0023</u>
	Ps	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	47.69	**54.23**	<u>49.34</u>	47.74	42.38	**42.84**	42.39	<u>42.55</u>	46.04	**56.00**	46.05	<u>46.49</u>
	MSSIM	0.9885	**0.9988**	<u>0.9937</u>	0.9886	0.9891	**0.9911**	0.9891	<u>0.9897</u>	0.9810	**0.9997**	0.9822	<u>0.9834</u>
Case 2	Setup	$\lambda_1 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	20.55	**21.64**	20.19	<u>20.84</u>	17.15	**17.50**	17.23	<u>17.38</u>	21.95	**28.67**	22.34	<u>22.72</u>
	RMSE	0.0234	**0.0206**	0.0240	<u>0.0226</u>	0.0310	**0.0297**	0.0308	<u>0.0302</u>	0.0043	**0.0020**	0.0041	<u>0.0040</u>
	Ps	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	41.32	**49.20**	<u>43.39</u>	41.70	38.29	**39.00**	38.35	<u>38.51</u>	40.18	**52.34**	40.45	<u>42.56</u>
	MSSIM	0.9530	**0.9959**	<u>0.9735</u>	0.9572	0.9726	**0.9787**	0.9731	<u>0.9743</u>	0.9321	**0.9991**	0.9361	<u>0.9624</u>
Case 3	Setup	$\lambda_1 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	<u>23.75</u>	**23.92**	21.06	23.73	20.53	**20.80**	20.46	<u>20.71</u>	26.33	**29.66**	25.91	<u>26.93</u>
	RMSE	<u>0.0162</u>	**0.0159**	0.0218	<u>0.0162</u>	0.0211	**0.0204**	0.0213	<u>0.0207</u>	0.0026	**0.0018**	0.0028	<u>0.0025</u>
	Ps	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	46.76	**50.54**	<u>48.43</u>	46.82	41.90	**42.24**	41.90	<u>42.06</u>	45.07	**55.27**	45.23	<u>47.14</u>
	MSSIM	0.9858	**0.9949**	<u>0.9922</u>	0.9860	0.9883	**0.9902**	0.9883	<u>0.9889</u>	0.9761	**0.9996**	0.9783	<u>0.9870</u>
Case 4	Setup	$\lambda_1 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	23.27	**23.59**	21.89	<u>23.31</u>	18.62	**19.02**	18.65	<u>18.92</u>	25.52	**30.92**	25.52	<u>26.68</u>
	RMSE	<u>0.0172</u>	**0.0166**	0.0199	<u>0.0172</u>	0.0263	**0.0251**	0.0262	<u>0.0254</u>	0.0029	**0.0016**	0.0029	<u>0.0025</u>
	Ps	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	45.24	**47.92**	<u>47.58</u>	45.63	40.78	**41.30**	40.80	<u>40.97</u>	43.65	**55.21**	44.34	<u>45.62</u>
	MSSIM	0.9790	<u>0.9893</u>	**0.9900**	0.9810	0.9843	**0.9871**	0.9844	<u>0.9852</u>	0.9665	**0.9996**	0.9730	<u>0.9803</u>
Case 5	Setup	$\lambda_1 = 10^{1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	19.96	**21.38**	<u>21.02</u>	19.96	21.30	**21.52**	21.27	<u>21.42</u>	21.22	**24.96**	21.46	<u>21.54</u>
	RMSE	0.0244	**0.0211**	<u>0.0218</u>	0.0244	0.0194	**0.0189**	0.0195	<u>0.0191</u>	0.0047	**0.0031**	0.0046	<u>0.0045</u>
	Ps	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	45.97	**50.17**	45.92	<u>45.99</u>	42.48	**42.83**	42.51	<u>42.65</u>	40.99	**47.52**	41.18	<u>41.51</u>
	MSSIM	<u>0.9893</u>	**0.9983**	0.9886	<u>0.9893</u>	0.9890	**0.9909**	0.9890	<u>0.9895</u>	0.9484	**0.9980**	0.9507	<u>0.9550</u>
Case 6	Setup	$\lambda_1 = 10^{0}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	16.28	**17.14**	14.86	<u>16.80</u>	<u>17.82</u>	**17.96**	17.55	17.81	19.98	**24.39**	<u>20.38</u>	20.36
	RMSE	0.0374	**0.0339**	0.0425	<u>0.0350</u>	<u>0.0288</u>	**0.0282**	0.0296	<u>0.0288</u>	0.0054	**0.0033**	<u>0.0051</u>	<u>0.0051</u>
	Ps	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	38.88	41.22	**42.52**	<u>42.03</u>	38.21	**38.55**	38.14	<u>38.24</u>	37.98	**47.02**	38.23	<u>38.86</u>
	MSSIM	0.9229	0.9560	**0.9741**	<u>0.9660</u>	0.9729	**0.9798**	0.9724	<u>0.9732</u>	0.8974	**0.9956**	0.9029	<u>0.9176</u>
Case 7	Setup	$\lambda_1 = 10^{1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{1}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$
	SRE	17.47	**17.90**	16.66	<u>17.48</u>	15.89	<u>15.95</u>	15.68	**16.05**	19.53	**25.57**	<u>20.01</u>	20.00
	RMSE	<u>0.0323</u>	**0.0307**	0.0351	<u>0.0323</u>	0.0356	<u>0.0353</u>	0.0362	**0.0348**	0.0056	**0.0029**	<u>0.0054</u>	<u>0.0054</u>
	Ps	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	39.68	**47.34**	39.58	<u>39.78</u>	35.80	<u>35.97</u>	35.77	**36.11**	36.85	**48.64**	37.09	<u>38.06</u>
	MSSIM	0.9392	**0.9957**	0.9378	<u>0.9406</u>	0.9533	<u>0.9553</u>	0.9531	**0.9570**	0.8653	**0.9969**	0.8716	<u>0.8980</u>
Case 8	Setup	$\lambda_1 = 10^{0}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{0}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{0}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	13.84	**14.96**	11.89	<u>14.08</u>	<u>14.63</u>	**14.67**	14.26	14.60	18.29	**21.12**	18.67	<u>18.69</u>
	RMSE	0.0485	**0.0427**	0.0592	<u>0.0468</u>	<u>0.0403</u>	**0.0401**	0.0415	0.0405	0.0064	**0.0047**	<u>0.0062</u>	<u>0.0062</u>
	Ps	**1.00**	**1.00**	**1.00**	**1.00**	**0.99**	**0.99**	**0.99**	**0.99**	**1.00**	**1.00**	**1.00**	**1.00**
	MPSNR	36.73	<u>40.36</u>	37.76	**41.30**	35.17	**35.40**	35.12	<u>35.21</u>	35.34	**43.66**	35.56	<u>36.44</u>
	MSSIM	0.8846	<u>0.9518</u>	0.9118	**0.9653**	0.9519	**0.9556**	0.9517	<u>0.9525</u>	0.8303	**0.9925**	0.8375	<u>0.8696</u>

The bold values are best values.

the variables updated by P-PDS approach the solution of our constrained convex optimization.

G. Ablation Experiments

To demonstrate the effectiveness of the image-domain regularization [the third term of (15)], we compared RHUIDR performance with the performance when the image-domain regularization was removed [referred to as RHUIDR (–)]. The hyperparameters λ_1, λ_3, ε, and η were set to the same as in RHUIDR.

Tables XIII and XIV show the SRE, RMSE, Ps, MPSNR, and MSSIM results of the ablation experiments for the synthetic and real datasets, respectively. The best and second-best results are highlighted in bold and underlined, respectively. RHUIDR with the image-domain regularization was superior to RHUIDR without the image-domain regularization. In particular, the image-domain regularization contributed to an improvement in SRE, RMSE, and Ps and a significant improvement in MPSNR and MSSIM. This implies that the reconstructed HS image has the desirable spatio-spectral property, resulting in the estimation of more appropriate abundance maps.

TABLE XIV
SRE, RMSE, PS, MPSNR, AND MSSIM OF THE ABLATION EXPERIMENTS USING REAL DATASETS

Image	Metrics	*Jasper Ridge*				*Samson*				*Urban*			
		RHUIDR –	**RHUIDR (HTV)**	**RHUIDR (SSTV)**	**RHUIDR (HSSTV)**	RHUIDR –	**RHUIDR (HTV)**	**RHUIDR (SSTV)**	**RHUIDR (HSSTV)**	RHUIDR –	**RHUIDR (HTV)**	**RHUIDR (SSTV)**	**RHUIDR (HSSTV)**
Case 1	Setup	$\lambda_1 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^1$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$
	SRE	19.08	19.07	**19.33**	19.06	15.90	**19.51**	16.48	18.03	15.00	**15.92**	14.49	14.74
	RMSE	0.0294	0.0294	**0.0286**	0.0295	0.0368	**0.0254**	0.0345	0.0296	0.0086	**0.0078**	0.0090	0.0088
	Ps	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**0.99**	**0.99**	**0.99**	**0.99**
	MPSNR	45.44	45.45	**45.47**	45.45	45.27	43.75	45.52	**46.74**	41.84	37.93	41.74	**41.91**
	MSSIM	**0.9882**	**0.9882**	0.9881	**0.9882**	0.9842	0.9836	0.9855	**0.9907**	0.9857	0.9741	0.9854	**0.9863**
Case 2	Setup	$\lambda_1 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^1$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$
	SRE	15.38	15.49	**15.53**	15.37	11.02	**14.51**	11.09	12.86	11.36	**12.62**	10.90	11.37
	RMSE	0.0441	0.0436	**0.0432**	0.0441	0.0597	**0.0435**	0.0590	0.0504	0.0126	**0.0110**	0.0131	0.0126
	Ps	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**	**0.98**	**0.98**	**0.98**	**0.98**
	MPSNR	40.34	**40.97**	40.31	40.36	40.10	**40.84**	40.45	40.44	35.89	35.25	35.77	**36.05**
	MSSIM	0.9639	**0.9732**	0.9634	0.9641	0.9513	0.9719	0.9561	**0.9725**	0.9459	**0.9571**	0.9448	0.9483
Case 3	Setup	$\lambda_1 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^1$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$
	SRE	18.76	18.76	**18.98**	18.75	15.24	**18.16**	15.61	16.74	14.33	**15.29**	13.79	14.11
	RMSE	0.0303	0.0304	**0.0296**	0.0304	0.0393	**0.0294**	0.0378	0.0341	0.0092	**0.0083**	0.0097	0.0094
	Ps	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**0.99**	**0.99**	**0.99**	**0.99**
	MPSNR	44.46	**44.47**	**44.47**	**44.47**	44.12	43.09	44.34	**45.45**	40.89	37.38	40.78	**40.98**
	MSSIM	0.9849	0.9849	0.9848	**0.9850**	0.9791	0.9812	0.9807	**0.9868**	0.9823	0.9712	0.9819	**0.9831**
Case 4	Setup	$\lambda_1 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^1$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$
	SRE	18.26	**18.31**	18.29	18.26	14.09	**16.09**	14.57	15.24	14.22	**14.95**	13.76	14.05
	RMSE	0.0321	**0.0319**	0.0321	0.0321	0.0439	**0.0364**	0.0417	0.0390	0.0093	**0.0086**	0.0097	0.0095
	Ps	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**0.99**	**0.99**	**0.99**	**0.99**
	MPSNR	43.37	**43.89**	43.39	43.39	43.18	42.80	43.41	**44.25**	40.00	37.34	39.87	**40.12**
	MSSIM	0.9801	**0.9836**	0.9803	0.9802	0.9742	0.9805	0.9761	**0.9815**	0.9782	0.9713	0.9776	**0.9792**
Case 5	Setup	$\lambda_1 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^1$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$
	SRE	16.19	**16.36**	16.16	16.19	10.75	**12.91**	10.04	12.12	9.61	**10.38**	9.59	9.59
	RMSE	0.0401	**0.0394**	0.0401	0.0400	0.0607	**0.0489**	0.0648	0.0532	0.0148	**0.0138**	0.0149	0.0148
	Ps	**1.00**	0.99	0.99	**1.00**	0.99	**1.00**	0.99	**1.00**	0.94	**0.96**	0.94	0.94
	MPSNR	41.47	**42.26**	41.63	41.49	41.05	**42.74**	40.95	42.48	37.42	30.36	37.42	**37.43**
	MSSIM	0.9734	**0.9791**	0.9744	0.9735	0.9631	**0.9811**	0.9625	0.9759	0.9677	0.8906	0.9677	**0.9679**
Case 6	Setup	$\lambda_1 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^1$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$
	SRE	14.17	**14.39**	14.03	14.19	9.92	**12.49**	9.33	11.97	8.13	**10.02**	8.59	8.69
	RMSE	0.0500	**0.0489**	0.0507	0.0500	0.0655	**0.0525**	0.0689	0.0543	0.0175	**0.0144**	0.0166	0.0165
	Ps	**0.99**	**0.99**	**0.99**	**0.99**	0.99	0.99	0.98	**1.00**	0.90	**0.95**	0.93	0.93
	MPSNR	37.86	**38.82**	37.78	38.04	37.76	38.41	37.76	**39.70**	33.25	**34.34**	33.87	34.18
	MSSIM	0.9407	**0.9563**	0.9396	0.9435	0.9219	0.9559	0.9222	**0.9573**	0.9109	**0.9496**	0.9227	0.9294
Case 7	Setup	$\lambda_1 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$
	SRE	13.34	**13.67**	13.27	13.35	8.26	**13.61**	8.38	11.28	8.12	**10.55**	7.97	8.32
	RMSE	0.0548	**0.0531**	0.0551	0.0547	0.0767	**0.0466**	0.0759	0.0588	0.0173	**0.0136**	0.0177	0.0171
	Ps	**0.98**	**0.98**	**0.98**	**0.98**	0.95	**1.00**	0.98	0.99	0.91	**0.96**	0.91	0.92
	MPSNR	37.02	**38.22**	36.94	37.25	36.88	**39.74**	37.62	38.83	**33.51**	32.38	32.70	33.19
	MSSIM	0.9265	**0.9502**	0.9250	0.9310	0.9036	**0.9651**	0.9317	0.9612	0.9141	**0.9250**	0.8999	0.9117
Case 8	Setup	$\lambda_1 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^0$, $\lambda_2 = 10^1$, $\varepsilon = 0.95$	$\lambda_1 = 10^0$, $\lambda_2 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.98$	$\lambda_1 = 10^{-1}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^0$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$	$\lambda_1 = 10^{-1}$, $\lambda_2 = 10^{-2}$, $\varepsilon = 0.95$
	SRE	12.20	**12.46**	12.12	12.21	7.43	**10.39**	6.26	9.40	7.25	**8.92**	6.64	7.32
	RMSE	0.0621	**0.0604**	0.0623	0.0619	0.0827	**0.0630**	0.0922	0.0714	0.0188	**0.0160**	0.0199	0.0186
	Ps	**0.97**	**0.97**	**0.97**	**0.97**	0.90	**0.99**	0.84	0.96	0.88	**0.92**	0.85	0.88
	MPSNR	35.74	**36.93**	35.67	35.97	35.82	37.22	35.18	**37.76**	31.41	**31.82**	31.34	31.62
	MSSIM	0.9067	**0.9371**	0.9055	0.9124	0.8875	**0.9447**	0.8707	0.9405	0.8736	**0.9192**	0.8727	0.8796

The bold values are best values.

H. Summary

We summarize the experimental discussion as follows.

1) From the results of experiments in Cases 1–4, and 7 and the ablation experiments, we see that image-domain regularizations improve the unmixing performance.
2) The results of experiments in Cases 5, 6, and 8 verify that RHUIDR accurately estimates abundance maps if HS images are degraded by various types of noise.
3) RHUIDR achieves good unmixing performance in experiments using both synthetic and real HS images.

V. CONCLUSION

In this article, we have proposed a new method for noise-robust unmixing. RHUIDR adopts the image-domain regularization and explicitly models three types of noises. We have formulated the unmixing problem as a constrained convex optimization problem that includes the regularization, and have developed the optimization algorithm based on P-PDS. Experiments on synthetic and real HS images have demonstrated the superiority of RHUIDR over existing methods. RHUIDR will have strong impacts on the field of remote sensing, including the estimation of abundance maps from HS images taken in

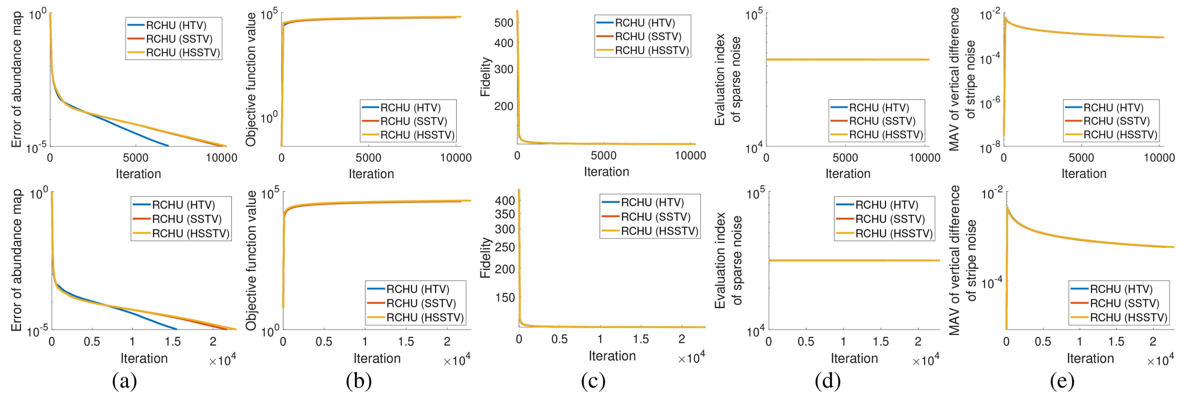

Fig. 17. Convergence analysis using the real images. The top row shows the results of experiments using *Jasper Ridge*. The bottom row shows the results of experiments using *Samson*. (a) Relative error of abundance maps $\|\mathbf{A}^{(t+1)} - \mathbf{A}^{(t)}\|_F / \|\mathbf{A}^{(t+1)}\|_F$ versus iteration t. (b) Objective function value $\|\mathbf{A}^{(t)}\|_{1,2,r} + \lambda_1\|\mathbf{D}(\mathbf{A}^{(t)})\|_1 + \lambda_2\mathcal{R}(\mathbf{K}(\mathbf{E}\mathbf{A}^{(t)})) + \lambda_3\|\mathbf{L}^{(t)}\|_1$ versus iteration t. (c) ℓ_2 distance between $\mathbf{V}$ and $\mathbf{E}\mathbf{A}^{(t)} + \mathbf{S}^{(t)} + \mathbf{L}^{(t)}$ versus iteration t. (d) ℓ_1 norm of $\mathbf{S}^{(t)}$ versus iteration t. (e) MAV of $\mathbf{D}_v(\mathbf{L}^{(t)})$ versus iteration t.

measurement environments with severe degradation. For future work, we will combine RHUIDR with a learning-based and a coarse abundance map-based weighted sparse unmixing approach to realize more noise-robust blind unmixing.

Acknowledgment

The authors would like to thank Ms. Yuki Nagamatsu, who was a student of Tokyo Institute of Technology, for her cooperation in the experiments.

References

[1] P. Ghamisi et al., "Advances in hyperspectral image and signal processing: A comprehensive overview of the state of the art," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 37–78, Dec. 2017.
[2] C.-I. Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*, vol. 1. New York, NY, USA: Springer Science & Business Media, 2003.
[3] P. S. Thenkabail and J. G. Lyon, *Hyperspectral Remote Sensing of Vegetation*. Boca Raton, FL, USA: CRC Press, 2016.
[4] B. Lu, P. D. Dao, J. Liu, Y. He, and J. Shang, "Recent advances of hyperspectral imaging technology and applications in agriculture," *Remote Sens.*, vol. 12, no. 16, 2020, Art. no. 2659.
[5] N. Keshava and J. F. Mustard, "Spectral unmixing," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 44–57, Jan. 2002.
[6] J. M. Bioucas-Dias et al., "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth. Observ. Remote Sens.*, vol. 5, no. 2, pp. 354–379, Apr. 2012.
[7] W.-K. Ma et al., "A signal processing perspective on hyperspectral unmixing: Insights from remote sensing," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 67–81, Jan. 2014.
[8] B. Rasti, P. Scheunders, P. Ghamisi, G. Licciardi, and J. Chanussot, "Noise reduction in hyperspectral imagery: Overview and application," *Remote Sens.*, vol. 10, no. 3, 2018, Art. no. 482.
[9] Y.-Q. Zhao and J. Yang, "Hyperspectral image denoising via sparse representation and low-rank constraint," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 1, pp. 296–308, Jan. 2015.
[10] G. Licciardi et al., "Decision fusion for the classification of hyperspectral data: Outcome of the 2008 GRS-S data fusion contest," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 11, pp. 3857–3865, Nov. 2009.
[11] N. Yokoya, C. Grohnfeldt, and J. Chanussot, "Hyperspectral and multispectral data fusion: A comparative review of the recent literature," *IEEE Trans. Geosci. Remote Sens. Mag.*, vol. 5, no. 2, pp. 29–56, Jun. 2017.
[12] M.-D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Sparse unmixing of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 6, pp. 2014–2039, Jun. 2011.
[13] M.-D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Collaborative sparse regression for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 1, pp. 341–354, Jan. 2014.
[14] H. Li, R. Feng, L. Wang, Y. Zhong, and L. Zhang, "Superpixel-based reweighted low-rank and total variation sparse unmixing for hyperspectral remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 629–647, Jan. 2021.
[15] L. Ren, Z. Ma, F. Bovolo, and L. Bruzzone, "A nonconvex framework for sparse unmixing incorporating the group structure of the spectral library," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2021, Art. no. 5506719.
[16] X. Shen, H. Liu, X. Zhang, K. Qin, and X. Zhou, "Superpixel-guided local sparsity prior for hyperspectral sparse regression unmixing," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6015105.
[17] Y. Liang, H. Zheng, G. Yang, Q. Du, and H. Su, "Superpixel-based weighted sparse regression and spectral similarity constrained for hyperspectral unmixing," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 6825–6842, 2023.
[18] M.-D. Iordache, J. Bioucas-Dias, and A. Plaza, "Total variation spatial regularization for sparse hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 11, pp. 4484–4502, Nov. 2012.
[19] H. K. Aggarwal and A. Majumdar, "Hyperspectral unmixing in the presence of mixed noise using joint-sparsity and total variation," *IEEE J. Sel. Topics Appl. Earth. Observ. Remote Sens.*, vol. 9, no. 9, pp. 4257–4266, Sep. 2016.
[20] R. Wang, H. Li, A. Pizurica, J. Li, A. Plaza, and W. Emery, "Hyperspectral unmixing using double reweighted sparse regression and total variation," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 7, pp. 1146–1150, Jul. 2017.
[21] S. Zhang, J. Li, C. Li, H. C. Deng, and A. Plaza, "Spectral–spatial weighted sparse regression for hyperspectral image unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3265–3276, Jun. 2018.
[22] J. Wang, T. Huang, J. Huang, H. Dou, L. Deng, and X. Zhao, "Row-sparsity spectral unmixing via total variation," *IEEE J. Sel. Topics Appl. Earth. Observ. Remote Sens.*, vol. 12, no. 12, pp. 5009–5022, Dec. 2019.
[23] F. Li, S. Zhang, C. Deng, B. Liang, J. Cao, and S. Wang, "Robust double spatial regularization sparse hyperspectral unmixing," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 12569–12582, 2021.
[24] T. Ince, "Double spatial graph Laplacian regularization for sparse unmixing," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 8009605.
[25] C. Deng et al., "Robust dual spatial weighted sparse unmixing for remotely sensed hyperspectral imagery," *Remote Sens.*, vol. 15, no. 16, 2023, Art. no. 4056.
[26] M. Zhao, X. Wang, J. Chen, and W. Chen, "A plug-and-play priors framework for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5501213.
[27] B. Rasti and B. Koirala, "SUnCNN: Sparse unmixing using unsupervised convolutional neural network," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 5508205.
[28] B. Rasti, B. Koirala, P. Scheunders, and P. Ghamisi, "UnDIP: Hyperspectral unmixing using deep image prior," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5504615.

[29] J. Huang, T.-Z. Huang, L.-J. Deng, and X.-L. Zhao, "Joint-sparse-blocks and low-rank representation for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 2419–2438, Apr. 2019.

[30] J. Huang, W.-C. Di, J.-J. Wang, J. Lin, and T.-Z. Huang, "Bilateral joint-sparse regression for hyperspectral unmixing," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 10147–10161, 2021.

[31] L. Wu, J. Huang, and M.-S. Guo, "Multidimensional low-rank representation for sparse hyperspectral unmixing," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, 2023, Art. no. 5502805.

[32] X.-R. Feng, H.-C. Li, J. Li, Q. Du, A. Plaza, and W. J. Emery, "Hyperspectral unmixing using sparsity-constrained deep nonnegative matrix factorization with total variation," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 10, pp. 6245–6257, Oct. 2018.

[33] H.-C. Li, X.-R. Feng, D.-H. Zhai, Q. Du, and A. Plaza, "Self-supervised robust deep matrix factorization for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5513214.

[34] X.-R. Feng, H.-C. Li, R. Wang, Q. Du, X. Jia, and A. Plaza, "Hyperspectral unmixing based on nonnegative matrix factorization: A comprehensive review," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 4414–4436, 2022.

[35] S. Ozkan, B. Kaya, and G. B. Akar, "EndNet: Sparse autoencoder network for endmember extraction and hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 482–496, Jan. 2019.

[36] B. Rasti, B. Koirala, P. Scheunders, and J. Chanussot, "MiSiCNet: Minimum simplex convolutional network for deep hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5522815.

[37] D. Hong et al., "Endmember-guided unmixing network (EGU-Net): A general deep learning framework for self-supervised hyperspectral unmixing," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 11, pp. 6518–6531, Nov. 2022.

[38] Q. Yuan, L. Zhang, and H. Shen, "Hyperspectral image denoising employing a spectral–spatial adaptive total variation model," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 10, pp. 3660–3677, Oct. 2012.

[39] W. Zhang, H. He, L. Zhang, H. Shen, and Q. Yuan, "Hyperspectral image restoration using low-rank matrix recovery," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 8, pp. 4729–4743, Aug. 2014.

[40] W. He, H. Zhang, L. Zhang, and H. Shen, "Total-variation-regularized low-rank matrix factorization for hyperspectral image restoration," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 178–188, Jan. 2016.

[41] H. K. Aggarwal and A. Majumdar, "Hyperspectral image denoising using spatio-spectral total variation," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 3, pp. 442–446, Mar. 2016.

[42] S. Takeyama, S. Ono, and I. Kumazawa, "A constrained convex optimization approach to hyperspectral image restoration with hybrid spatio-spectral regularization," *Remote Sens.*, vol. 12, no. 21, 2020, Art. no. 3541.

[43] M. Wang, Q. Wang, J. Chanussot, and D. Hong, "l_0-l_1 hybrid total variation regularization and its applications on hyperspectral image mixed noise removal and compressed sensing," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 9, pp. 7695–7710, Sep. 2021.

[44] S. Takemoto, K. Naganuma, and S. Ono, "Graph spatio-spectral total variation model for hyperspectral image denoising," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6012405.

[45] H. Carfantan and J. Idier, "Statistical linear destriping of satellite-based pushbroom-type images," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 4, pp. 1860–1871, Apr. 2010.

[46] X. Liu, H. Shen, Q. Yuan, X. Lu, and C. Zhou, "A universal destriping framework combining 1-D and 2-D variational optimization methods," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 808–822, Feb. 2018.

[47] K. Naganuma and S. Ono, "A general destriping framework for remote sensing images using flatness constraint," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5525016.

[48] T. Pock and A. Chambolle, "Diagonal preconditioning for first order primal-dual algorithms in convex optimization," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 1762–1769.

[49] K. Naganuma and S. Ono, "Variable-wise diagonal preconditioning for primal-dual splitting: Design and applications," *IEEE Trans. Signal Process.*, vol. 71, pp. 3281–3295, 2023.

[50] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems," *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 681–695, Mar. 2011.

[51] G. Chierchia, N. Pustelnik, J. Pesquet, and B. Pesquet-Popescu, "Epigraphical projection and proximal tools for solving constrained convex optimization problems," *Signal, Image Video Process.*, vol. 9, no. 8, pp. 1737–1749, 2015.

[52] S. Ono and I. Yamada, "Signal recovery with certain involved convex data-fidelity constraints," *IEEE Trans. Signal Process.*, vol. 63, no. 22, pp. 6149–6163, Nov. 2015.

[53] S. Ono, "Primal-dual plug-and-play image restoration," *IEEE Signal Process. Lett.*, vol. 24, no. 8, pp. 1108–1112, Aug. 2017.

[54] S. Ono, "Efficient constrained signal reconstruction by randomized epigraphical projection," in *Proc. IEEE Int. Conf. Acoust, Speech Signal Process.*, 2019, pp. 4993–4997.

[55] L. Condat, "Fast projection onto the simplex and the l1 ball," *Math. Prog.*, vol. 158, no. 1, pp. 575–585, Jul. 2016.

[56] C. Rodarmel and J. Shan, "Principal component analysis for hyperspectral image classification," *Surv. Land Inf. Sci.*, vol. 62, no. 2, pp. 115–122, 2002.

[57] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

Kazuki Naganuma (Graduate Student Member, IEEE) received the B.E. degree from the Kanagawa Institute of Technology, Atsugi, Japan, and the M.E. degree from the Tokyo Institute of Technology, Yokohama, Japan, in 2020, both in information and computer sciences. He is currently working toward the Ph.D. degree with the Department of Computer Science, Tokyo Institute of Technology.

Since 2023, he has been a Research Fellow (DC2) with the Japan Society for the Promotion of Science, Tokyo, Japan, and a Researcher of ACT-X with the Japan Science and Technology Corporation, Tokyo, Japan, since 2023. His research interests include signal and image processing, remote sensing, and optimization theory.

Mr. Naganuma was the recipient of the Student Conference Paper Award from IEEE Signal Processing Society Japan Chapter in 2023.

Shunsuke Ono (Senior Member, IEEE) received the B.E. degree in computer science and the M.E. and Ph.D. degrees in communications and computer engineering from the Tokyo Institute of Technology, Yokohama, Japan, in 2010, 2012, and 2014, respectively.

From 2012 to 2014, he was a Research Fellow (DC1) with the Japan Society for the Promotion of Science. He is currently an Associate Professor with the Department of Computer Science, School of Computing, Tokyo Institute of Technology. From 2016 to 2020 and since 2021, he was/has been a Researcher with Precursory Research for Embryonic Science and Technology (PRESTO), Japan Science and Technology Agency, Tokyo, Japan. His research interests include signal processing, image analysis, remote sensing, mathematical optimization, and data science.

Dr. Ono was the recipient of the Young Researchers' Award and the Excellent Paper Award from the Institute of Electronics, Information and Communication Engineers in 2013 and 2014, respectively, the Outstanding Student Journal Paper Award and the Young Author Best Paper Award from the IEEE Signal Processing Society (SPS) Japan Chapter in 2014 and 2020, respectively, the Funai Research Award from the Funai Foundation in 2017, the Ando Incentive Prize from the Foundation of Ando Laboratory in 2021, the Young Scientists' Award from the Ministry of Education, Culture, Sports, Science and Technology in 2022, and the Outstanding Editorial Board Member Award from IEEE SPS in 2023. He has been an Associate Editor for IEEE TRANSACTIONS ON SIGNAL AND INFORMATION PROCESSING OVER NETWORKS, since 2019.