Futuristic Greedy Approach to Sparse Unmixing of Hyperspectral Data

Naveed Akhtar, Faisal Shafait, and Ajmal Mian

***Abstract*—Spectra measured at a single pixel of a remotely sensed hyperspectral image is usually a mixture of multiple spectral signatures (endmembers) corresponding to different materials on the ground. Sparse unmixing assumes that a mixed pixel is a sparse linear combination of different spectra already available in a spectral library. It uses sparse approximation (SA) techniques to solve the hyperspectral unmixing problem. Among these techniques, greedy algorithms suite well to sparse unmixing. However, their accuracy is immensely compromised by the high correlation of the spectra of different materials. This paper proposes a novel greedy algorithm, called OMP-Star, that shows robustness against the high correlation of spectral signatures. We preprocess the signals with spectral derivatives before they are used by the algorithm. To approximate the mixed pixel spectra, the algorithm employs a futuristic greedy approach that, if necessary, considers its future iterations before identifying an endmember. We also extend OMP-Star to exploit the nonnegativity of spectral mixing. Experiments on simulated and real hyperspectral data show that the proposed algorithms outperform the state-of-the-art greedy algorithms. Moreover, the proposed approach achieves results comparable to convex relaxation-based SA techniques, while maintaining the advantages of greedy approaches.**

***Index Terms*—Greedy algorithm, hyperspectral unmixing, orthogonal matching pursuit (OMP), sparse unmixing.**

I. Introduction

HYPERSPECTRAL remote sensing extracts information from the scenes on the Earth's surface, using the radiance measured by airborne or spaceborne sensors [1], [2]. These sensors measure the spectra of the Earth's surface at hundreds of contiguous narrow bands [3], resulting in a hyperspectral data cube that has two spatial and one spectral dimension (see Fig. 1). Each pixel of such a data cube is a vector that represents the spectral signature of the objects/materials measured at the pixel. Due to low spatial resolution of sensors, presence of intimate mixtures of materials, and multiple scattering, the signature of the pixel is usually a combination of several pure spectral signatures. Each of these pure spectral signatures is called an *endmember*. Hyperspectral unmixing aims at extracting these endmembers and their *fractional abundances* (i.e., proportion of the endmembers in a pixel), one set per pixel [4].

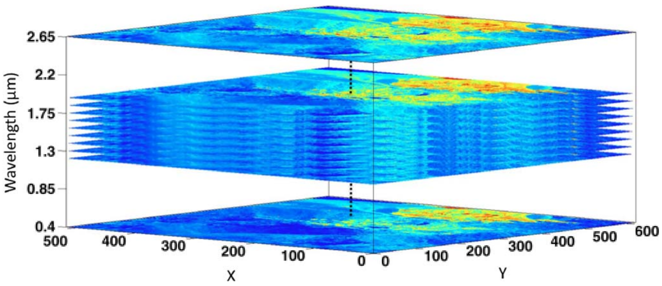


Fig. 1. Illustration of hyperspectral data cube: The xy plane corresponds to spatial dimensions. The spectral dimension shows the data collected at different wavelengths. The cube is illustrated with only eleven spectral bands (out of 224). The data are taken over Cuprite mines, Nevada, by AVIRIS [29].

In recent years, linear unmixing of hyperspectral data has attracted significant interest of researchers [1]. Linear unmixing assumes that mixed spectra in a hyperspectral data cube can be expressed as a linear combination of the endmembers, which are weighted by their fractional abundances. Many works under this model have exploited the geometric properties of the hyperspectral data (e.g., [5]–[8]). Such approaches exploit the fact that the convex hull of pure endmembers in the data forms a probability simplex. Thus, finding the endmembers amounts to finding the vertices of the simplex [9], [10]. Most of the classical geometrical methods for unmixing assume the presence of at least one pure pixel for every material captured in the scene. Vertex component analysis [12], pixel purity index [17], simplex growing algorithm [7], successive volume maximization [13], N-FINDER [14], iterative error analysis [15], and recursive algorithm for separable nonnegative matrix factorization [16] are popular examples of such methods.

The assumption of the existence of pure spectra in a hyperspectral data cube is not a practical one. Therefore, techniques, such as minimum volume simplex analysis [18], minimum volume transform-nonnegative matrix factorization [19], iterative constrained endmembers (ICE) [20], and sparsity-promoting ICE [21], have been proposed to circumvent the problem by generating pure endmembers from the hyperspectral data themselves. However, these techniques are likely to fail in highly mixed scenarios, where the algorithms end up generating artificial endmembers that cannot be associated with the spectral signatures of true materials [22]. For such cases, hyperspectral unmixing is usually formulated as a statistical inferencing problem, and the Bayesian paradigm becomes the common choice [1]. Under this paradigm, the computational complexity of Bayesian inferencing becomes a bottleneck for effective hyperspectral unmixing.

Manuscript received December 10, 2013; revised April 17, 2014 and June 19, 2014; accepted August 10, 2014. This work was supported by the Australian Research Council under Grant DP110102399.

The authors are with the School of Computer Science and Software Engineering, The University of Western Australia, Crawley, W.A. 6009, Australia (e-mail: naveed.akhtar@research.uwa.edu.au; faisal.shafait@uwa.edu.au; ajmal.mian@uwa.edu.au).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TGRS.2014.2356556

In order to overcome the aforementioned issues, hyperspectral unmixing has recently been approached in a semisupervised fashion [11]. This approach formulates hyperspectral unmixing as a sparse approximation (SA) problem and aims at developing efficient and accurate SA algorithms for sparse unmixing. Sparse unmixing makes use of a library of pure spectra and finds the optimal subset from the library that can best model the mixed pixel [4]. Iordache *et al.* [11] have studied different SA algorithms for sparse unmixing of hyperspectral data, including orthogonal matching pursuit (OMP) [23], basis pursuit (BP) [24], BP denoising (BPDN) [24] and iterative spectral mixture analysis [25]. Recently, the authors have also exploited the spatial information [27] and the subspace nature [26] of the hyperspectral data in sparse unmixing. Previous works in sparse unmixing have mainly focused on SA algorithms that are based on convex relaxation of the problem. Generally, the SA algorithms based on the greedy approach [23] have lower computational complexity than their convex relaxation counterparts [31], [32]. These algorithms find approximate solution for the l_0 problem directly, without smoothing the penalty function [32]. Furthermore, the greedy algorithms admit to simpler and faster implementation [45]. However, Iordache *et al.* [11] have shown that, in comparison to the convex relaxation algorithms, the accuracy of the greedy algorithms (e.g., OMP) is adversely affected by the high correlation between the spectra of the pure materials.

Shi *et al.* [32] have strongly argued to exploit the potential of the greedy approach for sparse unmixing and propose a greedy algorithm, which is called simultaneous matching pursuit (SMP). This algorithm processes the data, in terms of spatial blocks, and exploits the contextual information in the data to mitigate the problems caused by the high correlation among the spectra. However, the block size becomes an important parameter for SMP that is specific to hyperspectral data cube. Furthermore, the work assumes the existence of only a few endmembers in the whole data cube and the presence of spatial information everywhere in the data. Since hyperspectral unmixing is primarily a pixel-based process, we favor pixel-based greedy algorithms for sparse unmixing. These algorithms do not assume the existence of contextual information in the data. However, they can always be enhanced to take advantage of the contextual information following the guidelines in [47].

In this paper, we propose a greedy algorithm for sparse unmixing, which is called OMP-Star. OMP-Star is a pixel-based algorithm that augments OMP's greedy pursuit strategy with a futuristic heuristic. This heuristic is inspired by a popular search algorithm, which is called A-Star [43]. OMP-Star shows robustness against the problems caused by the high correlation among the spectra, while maintaining the advantages of greedy algorithms. We further modify the proposed algorithm, such that it takes advantage of the nonnegative nature of the fractional abundances. This constrained version of the proposed algorithm is called OMP-Star+. The second contribution of this paper is that it exploits derivatives [46] of hyperspectral data for sparse unmixing with *greedy algorithms*. It is possible to reduce the correlation among spectral signatures by taking their derivatives [48], [49]. Therefore, we preprocess hyperspectral data with derivatives for all the greedy algorithms. Although preprocessing the data reduces the correlation among the spectra, the reduction is generally not sufficient to achieve accurate results using the greedy algorithms. Therefore, a greedy algorithm with robustness against high signal correlation remains desirable for hyperspectral unmixing. This paper also makes a minor theoretical contribution to sparse unmixing by deriving the condition for spectral libraries, which ensures that the nonnegativity of fractional abundances automatically constrains their sum to constant values. This is important because once a spectral library is processed to satisfy this condition, we do not need to *explicitly* impose the aforementioned constraint, which must be satisfied by the computed fractional abundances. We test the proposed approach thoroughly on simulated and real hyperspectral data. The results show that the proposed algorithms show better performance than the existing greedy algorithms. Furthermore, the proposed approach achieves results comparable to the convex relaxation-based SA algorithms, with a considerable computational advantage.

This paper is organized as follows. Section II formulates hyperspectral unmixing as an SA problem and presents the aforementioned theoretical contribution of this paper. In Section III, we review the important greedy algorithms for SA. Section IV presents the proposed algorithms. We study derivatives for sparse unmixing in Section V. The proposed algorithms are evaluated with synthetic data in Section VI and with real data in Section VII. The computational complexity analysis of the proposed algorithms is presented in Section VIII. We draw conclusions in Section IX.

II. Hyperspectral Unmixing as SA

A. Linear Mixing Model

Hyperspectral unmixing as an SA problem focuses on a linear mixing model (LMM). The LMM assumes that the spectral response at a band in the mixed pixel is a linear combination of the constituent endmembers at that band. Written mathematically, we have

$$y_i = \sum_{j=1}^{p} l_{ij}\alpha_j + \epsilon_i \tag{1}$$

where y_i is the spectral reflectance measured at the ith band, p is the total number of endmembers in the mixed pixel, l_{ij} is the reflectance of the jth endmember at band i, α_j is the fractional abundance of the corresponding endmember, and ϵ_i is the noise affecting the measurement. Assuming that the hyperspectral data cube is acquired by a sensor with m spectral channels, the LMM can be written in a matrix form

$$\mathbf{y} = \mathbf{L}\boldsymbol{\alpha} + \boldsymbol{\epsilon} \tag{2}$$

where $\mathbf{y} \in \mathbb{R}^m$ represents the measured reflectance at a pixel, $\mathbf{L} \in \mathbb{R}^{m\times p}$ is a matrix containing the endmembers, $\boldsymbol{\alpha} \in \mathbb{R}^p$ is a vector with fractional abundances of the corresponding endmembers, and $\boldsymbol{\epsilon} \in \mathbb{R}^m$ represents noise.

In the LMM, fractional abundances of the endmembers must satisfy two constraints [1], i.e., 1) **ANC**: *Abundance Nonnegativity Constraint* $(\forall_i, i \in \{1,\ldots,p\}, \alpha_i > 0)$ and 2) **ASC**:

Abundance Sum-to-one Constraint ($\sum_{i=1}^{p} \alpha_i = 1$). These constraints owe to the fact that the fractional abundances of the endmembers are nonnegative quantities that sum up to 1 for each pixel

B. Sparse Unmixing

Let us denote a spectral library by a matrix $\mathbf{D} \in \mathbb{R}^{m \times k} (k > m)$, with each column $\mathbf{d}_i \in \mathbb{R}^m$ representing the spectral signature of a pure material. Under the assumption that $\mathbf{D}$ includes all the endmembers of a pixel, the signal $\mathbf{y}$ at the pixel can be reconstructed as

$$\mathbf{y} = \mathbf{D}\boldsymbol{\alpha} + \boldsymbol{\epsilon} \tag{3}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^k$ has only p nonzero elements ($p \ll k$). Without considering the aforementioned constraints over the fractional abundances, sparse unmixing is formulated as the following energy optimization problem:

$$(P_0^{\eta}) : \quad \min_{\alpha} \|\boldsymbol{\alpha}\|_0 \quad \text{s.t.} \quad \|\mathbf{D}\boldsymbol{\alpha} - \mathbf{y}\|_2 \leq \eta \tag{4}$$

where $\|.\|_0$ is the l_0-pseudonorm that simply counts the number of nonzero elements in $\boldsymbol{\alpha}$, and η is the tolerance due to noise and modeling error.

(P_0^{η}) is generally an NP-hard problem [30], and in practice, it is usually solved with the greedy SA algorithms (e.g., OMP) or by convexification of the problem (e.g., BP). In sparse unmixing, the support of the solution $\boldsymbol{\alpha}$ denotes the indexes of the endmembers in the spectral library $\mathbf{D}$. Note that the goal of hyperspectral unmixing is to find 1) the correct set of endmembers and 2) their corresponding fractional abundances. This is different from the goal of high-fidelity SA of $\mathbf{y}$ using $\mathbf{D}$. In fact, with similar spectral signatures in $\mathbf{D}$, a good solution to the latter may be practically useless for the former.

Introducing the ANC in (P_0^{η}) gives us the following formulation:

$$\left(P_0^{\eta+}\right) : \quad \min_{\alpha} \|\boldsymbol{\alpha}\|_0 \quad \text{s.t.} \quad \|\mathbf{D}\boldsymbol{\alpha} - \mathbf{y}\|_2 \leq \eta, \quad \boldsymbol{\alpha} > \mathbf{0} \tag{5}$$

$(P_0^{\eta+})$ is constrained by ANC and not by ASC. Previous works in sparse unmixing (e.g., [11] and [32]) state that imposing ANC in sparse unmixing automatically imposes a general version of ASC on the problem. This may not *always* be the case; however, if $\mathbf{D}$ is normalized in l_1-norm, then the statement is true in general.[1] We augment this argument with analytical results presented in Section II-C. The general version of ASC implies, for a sparse unmixing solution $\boldsymbol{\alpha}$, $\|\boldsymbol{\alpha}\|_1 = c$ [11]. Here, $\|.\|_1$ represents the l_1-norm of the vector, and c is a pixel-dependent scale factor. The generalized ASC simply becomes ASC when $c = 1$.

Minimization of the l_0-pseudonorm of $\boldsymbol{\alpha}$ is generally performed with greedy SA algorithms. Relaxed convexification of this minimization problem has also been widely researched. With relaxed convexification, the sparse unmixing problem can be rewritten as

$$\left(P_1^{\eta+}\right) : \quad \min_{\alpha} \|\boldsymbol{\alpha}\|_1 \quad \text{s.t.} \quad \|\mathbf{D}\boldsymbol{\alpha} - \mathbf{y}\|_2 \leq \eta, \quad \boldsymbol{\alpha} > \mathbf{0} \tag{6}$$

$(P_1^{\eta+})$ is more tractable than $(P_0^{\eta+})$ because of the convex nature of l_1-norm [33]. In the context of sparse unmixing, Constraint Spectral Unmixing by variable Splitting and Augmented Langrangian (CSUnSAL) [34] is a popular algorithm for solving $(P_1^{\eta+})$. CSUnSAL exploits the Alternating Direction Method of Multipliers (ADMM) presented in [35], and it is tailored for hyperspectral unmixing.

If we neglect the ANC in $(P_1^{\eta+})$, then the rest of the problem is equivalent to the well-known least absolute shrinkage and selection operator (LASSO) [36] problem, given below as (P_1^{λ}). Of course, we need an appropriate Langrangian multiplier λ for this equivalency. Hence

$$\left(P_1^{\lambda}\right) : \quad \min_{\alpha} \frac{1}{2}\|\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}\|_2 + \lambda\|\boldsymbol{\alpha}\|_1. \tag{7}$$

Iordache *et al.* [11] used Sparse Unmixing by variable Splitting and Augmented Langrangian (SUnSAL) [34] to solve (P_1^{λ}) for hyperspectral unmixing. Similar to CSUnSAL, SUnSAL exploits ADMM for sparse unmixing. Recent works [27], [28] in sparse unmixing, focusing on convex relaxation of the problem, have also exploited the piecewise smoothness of fractional abundances in hyperspectral images. More specifically, they add a total variation regularizer in $(P_1^{\eta+})$. Similarly, Iordache *et al.* [26] have also made use of the subspace nature of hyperspectral image for improved results via SUnSAL. In [26], SUnSAL is extended to collaborative SUnSAL (CLSUnSAL) that solves (P_1^{λ}) under a structured SA framework.

C. Automatic Imposition of ASC With ANC

Theorem 1: For a given spectral library $\mathbf{D} \in \mathbb{R}^{m \times k}$, if $\exists \mathbf{h} > \mathbf{0}$, such that $\mathbf{h}^T\mathbf{D} = l\mathbf{1}^T$ for $l > 0$, then $\boldsymbol{\alpha} \in \{\mathbf{y} = \mathbf{D}\boldsymbol{\alpha}, \boldsymbol{\alpha} > \mathbf{0}\}$ also satisfies $\|\boldsymbol{\alpha}\|_1 = c$, where $\mathbf{h} \in \mathbb{R}^{m \times 1}$, $\mathbf{1} \in \mathbb{R}^{k \times 1}$ is a vector of 1s, and c is a constant.

Proof: Consider the following model that admits to a sparse solution:

$$\mathbf{D}\boldsymbol{\alpha} = \mathbf{y}, \quad \boldsymbol{\alpha} > \mathbf{0}. \tag{8}$$

Given that $\mathbf{D}$ is a nonnegative matrix, $\exists \mathbf{h} \in \mathbb{R}^{m \times 1}$ such that

$$\mathbf{h}^T\mathbf{D}\boldsymbol{\alpha} = \mathbf{h}^T\mathbf{y} = v, \qquad \mathbf{h} > \mathbf{0}, \quad v > 0. \tag{9}$$

Let us impose that a valid solution of (8) also satisfies $\|\boldsymbol{\alpha}\|_1 = c$, where $c > 0$, then

$$l\mathbf{1}^T\boldsymbol{\alpha} = v, \qquad l > 0 \tag{10}$$

where $\mathbf{1} \in \mathbb{R}^{k \times 1}$ is a vector of 1s, and $l = v/c$. From (9) and (10), we arrive at

$$\mathbf{h}^T\mathbf{D} = l\mathbf{1}^T, \qquad l > 0. \tag{11}$$

[1] Here, we do not claim incorrectness of the previous works, but we emphasize the importance of normalization. Using $\mathbf{D}$ without normalizing the spectra may not automatically guarantee the imposition of the generalized ASC along with ANC. This fact has not been stated clearly in the previous literature related to *sparse unmixing*.

We can reach (11) only under the assumption that $\|\boldsymbol{\alpha}\|_1 = c$. Hence, for a given $\mathbf{D}$, if $\exists \mathbf{h}$ such that (11) is satisfied, a solution to (8) will also satisfy $\|\boldsymbol{\alpha}\|_1 = c$.

Corollary 1: If $\mathbf{d}_i$ represents the ith column of $\mathbf{D}$ and

$$\|\mathbf{d}_i\|_1 = l \; \forall i \in \{1, \ldots, k\}, \quad l > 0 \tag{12}$$

then (11) is always satisfied by $\mathbf{h} = \mathbf{1}$, where $\mathbf{1} \in \mathbb{R}^{m \times 1}$ is a vector of 1s.

Corollary 2: a) If the columns of $\mathbf{D}$ are normalized in l_1-norm, then $\|\boldsymbol{\alpha}\|_1 = c = \|\mathbf{y}\|_1$, i.e., automatic imposition of the generalized ASC due to ANC. Here, $\|\mathbf{y}\|_1$ is a pixel-dependent scaling factor. b) If the columns of $\mathbf{D}$ are normalized in l_1-norm and scaled by $\|\mathbf{y}\|_1$, then $\|\boldsymbol{\alpha}\|_1 = 1$, i.e., automatic imposition of ASC due to ANC.

The result in Corollary 2a follows from the reasoning: When $\mathbf{D}$ is normalized in l_1-norm, $l = 1$ in (12). Therefore, (11) is always satisfied by $\mathbf{h} = \mathbf{1}$ and $\mathbf{1}^T\mathbf{y} = \|\mathbf{y}\|_1 = v$ according to (9). This makes $c = v$ (since $l = v/c$) and $\mathbf{1}^T\boldsymbol{\alpha} = \|\boldsymbol{\alpha}\|_1 = c$ according to (10). Similarly, with $l = \|\mathbf{y}\|_1$ in (12), we finally get $\|\boldsymbol{\alpha}\|_1 = c = 1$. Note that, for a hyperspectral unmixing problem, generally $\nexists \mathbf{h}$ that satisfies (11). However, using the aforementioned results, we can always process $\mathbf{D}$ to ensure that the generalized ASC is automatically satisfied by a solution of (8). This processing requires simple l_1-normalization of the library according to Corollary 2. Previous works in sparse unmixing have mentioned Elad's results in [31] for automatic imposition of generalized ASC due to ANC. The aforementioned conclusions are in line with those results. The Appendix shows how we can extend Elad's results to arrive at the conclusion in Corollary 2a. However, the analysis presented here is more general, and the results shown here subsume those in the Appendix.

III. Greedy Algorithms

Greedy algorithms provide a polynomial time approximation of (P_0^η) by iteratively selecting the columns of $\mathbf{D}$ that best approximate $\mathbf{y}$. We first discuss OMP [23] as the representative algorithm, which is given in Algorithm 1. Each iteration of OMP can be divided into three steps. In the first step, OMP identifies the column in $\mathbf{D}$ that minimizes the residue of the current approximation of $\mathbf{y}$ (*identification*), where $\mathbf{y}$ itself is considered as the residual vector in the first iteration. The identified column is added to a selected subspace (*augmentation*). Next, a new residual vector is computed by approximating $\mathbf{y}$ in the column span of the selected subspace and subtracting this approximation from $\mathbf{y}$ (*residual update*). The aforementioned three steps are repeated until the *stopping rule* is satisfied. OMP updates the residual vector by computing the least squares approximation of $\mathbf{y}$ with the selected subspace, which makes the updated residual vector *orthogonal* to the selected subspace. Therefore, each newly identified column of $\mathbf{D}$ is different from those already present in the selected subspace. OMP's procedure of residual update is an enhancement over the matching pursuit (MP) algorithm [37] and also the reason why it is called the *Orthogonal* MP. The MP algorithm updates the residual vector by simply deflating it with the recently identified column of $\mathbf{D}$.

Algorithm 1 OMP

```
Initialization:
 1: Iteration: $i = 0$
 2: Initial solution: $\boldsymbol{\alpha}^0 = \mathbf{0}$
 3: Initial residual: $\mathbf{r}^0 = \mathbf{y} - \mathbf{D}\boldsymbol{\alpha}^0 = \mathbf{y}$
 4: Selected support: $\mathcal{S}^0 = support\{\boldsymbol{\alpha}^0\} = \emptyset$
Main Iteration: Update iteration: $i = i + 1$
    Identification
 5:     Compute $\epsilon(j) = \min_{z_j} \|\mathbf{d}_j z_j - \mathbf{r}^{i-1}\|_2^2, \forall j \in \{1, \ldots, k\}$,
        using the optimal choice $z_j^* = \mathbf{d}_\mathbf{j}^\mathbf{T}\mathbf{r}^{i-1}/\|\mathbf{d}_j\|_2^2$
 6:     Find a minimizer, $j_0$ of $\epsilon(j) : \forall j \notin \mathcal{S}^{i-1}, \epsilon(j_0) \leq \epsilon(j)$
    Augmentation:
 7:     $\mathcal{S}^i = \mathcal{S}^{i-1} \cup \{j_0\}$
    Residual update:
 8:     Compute $\boldsymbol{\alpha}^i = \min_{\boldsymbol{\alpha}} \|\mathbf{D}\boldsymbol{\alpha} - \mathbf{y}\|_2^2$ s.t. $support\{\boldsymbol{\alpha}^i\} = \mathcal{S}^i$
 9:     $\mathbf{r}^i = \mathbf{y} - \mathbf{D}\boldsymbol{\alpha}^i$
    Stopping rule:
10:     If $\|\mathbf{r}^i\|_2 < \epsilon_0$, stop. Otherwise iterate again.
```

Different enhancements over OMP have also been proposed in the literature related to image and signal processing. A nonnegative variant of OMP, henceforth denoted as OMP+, is proposed in [31]. OMP+ differs from OMP mainly in the *residual update* step, where the vector $\boldsymbol{\alpha}^i$ is constrained to have only positive elements in line "8" of Algorithm 1. Wang *et al.* [38] also proposed a generalized version of OMP, which is called generalized OMP (gOMP). The difference between OMP and gOMP is in the *identification* and *augmentation* steps. Instead of identifying a single column of $\mathbf{D}$, gOMP identifies L columns in each iteration and augments the selected subspace with all of these vectors. Here, L is an algorithm parameter. Lookahead OMP (LAOMP) [39] is a variant of OMP that modifies the *identification* step. LAOMP also identifies L (a prefixed number) columns in the *identification* step. Then, it picks one of these vectors and temporarily augments the selected subspace with it. It then proceeds similar to OMP until the *stopping rule* is satisfied. At the end, it stores the leftover residue. This procedure is repeated for all the L identified columns. LAOMP then permanently augments the selected subspace with the column that resulted in the least leftover residue, neglecting the other $L - 1$ vectors. A$*$OMP [51] is another variant of OMP that directly integrates the A-Star search [43] in OMP. We defer further discussion on this approach to Section IV, where we compare and contrast this approach with the proposed algorithm.

Subspace pursuit (SP) [42], compressive sampling MP (CoSaMP) [40], and regularized OMP (ROMP) [41] are the greedy algorithms that assume prior knowledge of the cardinality p of $\mathbf{y}$. All of these algorithms identify multiple columns of $\mathbf{D}$ in the *identification* step. In each iteration, SP identifies p columns and augments the selected subspace with all of them.

It then approximates $\mathbf{y}$ with the augmented subspace in the least squares sense. Then, it selects the vectors of the augmented subspace that correspond to the coefficients of the solution with p largest magnitudes. These vectors compose the updated selected subspace. Once the selected subspace is updated, SP updates the residual vector similar to OMP. CoSaMP identifies $2p$ columns in each iteration and augments the selected subspace with all of them. Then, it selects p vectors similar to SP. However, it updates the residual vector by using the already computed coefficients of the selected p vectors. ROMP also identifies p columns in each iteration. Then, it drops off some of these vectors using a predefined regularization rule before the *augmentation* step. In ROMP, the residual vector is also updated following OMP. The algorithms mentioned in this paragraph converge to solutions very quickly. However, the assumption of prior knowledge of the mixed signal's cardinality compromises their practicality for sparse unmixing.

IV. Proposed Algorithms

Here, we present the proposed greedy algorithm, which is called OMP-Star.[2] A nonnegative variant of this algorithm is developed later here. OMP-Star itself can be considered a variant of OMP, with its abilities enhanced following an approach inspired by A-Star [43], hence named OMP-Star. Before giving a detailed account on the proposed algorithms, let us briefly discuss the most significant drawback of the greedy algorithms, i.e., the problem of getting stuck in *local optima* of the solution space. This discussion will help in understanding the intuition behind the better performance of the proposed algorithms.

Consider the *identification* step of OMP, where it identifies the column $\mathbf{d}_{j_0}$ of $\mathbf{D}$, that minimizes the current residual. Assume another column of $\mathbf{D}$ ($\mathbf{d}_k : k \neq j_0$) that also causes *almost the same* amount of reduction in the residual, but it is not the minimum (because of $\mathbf{d}_{j_0}$). Being greedy, OMP prefers $\mathbf{d}_{j_0}$ over $\mathbf{d}_k$ for augmenting the selected subspace. Here, the question arises, should we simply neglect $\mathbf{d}_k$ and make the locally optimal choice of $\mathbf{d}_{j_0}$? What if, $\mathbf{d}_k$ was actually the right vector to be picked but $\mathbf{d}_{j_0}$ got selected only because of its similarity to $\mathbf{d}_k$? Such locally optimal choices of greedy algorithms trap them in local optima, i.e., a problem well researched for *greedy search algorithms* in the artificial intelligence literature [44]. In the context of greedy search algorithms, A-Star [43] is among the most effective approaches to solve the problem of local optimality of the solutions. The main idea behind this approach is to make selections in the search procedure keeping in view 1) the current benefits and 2) the future benefits of making a selection. This futuristic greedy approach helps in avoiding the local optima.

A. OMP-Star

OMP-Star uses a futuristic approach in the greedy pursuit strategy. The proposed algorithm initializes by considering the complete signal $\mathbf{y}$ as the residue. In the main iteration, it first identifies the column $\mathbf{d}_{j_0}$ of $\mathbf{D}$ that maximally correlates with the current residual vector. It uses this column to further identify L columns in $\mathbf{D}$ that best correlate with the residual vector. When normalized in l_2-norm, the inner product of each of these L vectors with the residual vector is at least a fraction t of the inner product of the unit vector of $\mathbf{d}_{j_0}$ and the residual vector (line "7" in Algorithm 2). Here, $t \in (0, 1)$ is an algorithm parameter. The indexes of thus identified $L + 1$ columns form a set $\mathcal{T}$. OMP-Star then points out the index $j^* \in \mathcal{T}$ of the column that it chooses to augment the selected subspace. For a moment, let us skip the discussion on the selection procedure of j^* (line "9" to "13" in Algorithm 2). Assume that j^* is exactly the index that we would like to select. This index is added to a set $\mathcal{S}^{i-1}$ (i denotes the current iteration) containing the indexes of $\mathbf{D}$'s columns that form the selected subspace. Once the selected subspace is augmented with the column corresponding to j^*, its span is used for updating the residual vector similar to OMP. This process is repeated until one of the stopping rules in line "17" of the algorithm is satisfied. Among these rules, (a) and (b) are obvious and well known. In the rule (c), we use a residual decay parameter $\beta \in (0, 1)$. This rule ensures that the algorithm stops if it is not able to reduce the l_2-norm of the residual vector at least by a fraction β in its last iteration.

Algorithm 2 OMP-Star

```
Initialization:
 1: Iteration: $i = 0$
 2: Initial solution: $\boldsymbol{\alpha}^0 = \mathbf{0}$
 3: Initial residual: $\mathbf{r}^0 = \mathbf{y} - \mathbf{D}\boldsymbol{\alpha}^0 = \mathbf{y}$
 4: Selected support: $\mathcal{S}^0 = support\{\boldsymbol{\alpha}^0\} = \emptyset$
Main Iteration: Update iteration: $i = i + 1$
    Identification:
 5:     Compute $\epsilon(j) = \min_{z_j} \|\mathbf{d}_j z_j - \mathbf{r}^{i-1}\|_2^2$, $\forall j \in \{1, \ldots, k\}$,
    using the optimal choice $z_j^* = \mathbf{d}_j^T \mathbf{r}^{i-1} / \|\mathbf{d}_j\|_2^2$
 6:     Find a minimizer, $j_0$ of $\epsilon(j) : \forall j \notin \mathcal{S}^{i-1}$, $\epsilon(j_0) \leq \epsilon(j)$
 7:     Find $j_1$ to $j_L : (\mathbf{d}_{j_l}^T \mathbf{r}^{i-1} / \|\mathbf{d}_{j_l}\|_2^2) \geq t \times (\mathbf{d}_{j_0}^T \mathbf{r}^{i-1} / \|\mathbf{d}_{j_0}\|_2^2)$
 8:     $\mathcal{T} = \{j_0, \ldots, j_L\}$
 9: if cardinality of $\mathcal{T} = 1$ then
10:         $j^* = j_0$
11: else
12:     $j^* \leftarrow ForwardSelection(\mathbf{D}, \mathbf{y}, \mathcal{S}^{i-1}, \mathcal{T}, f)$
13: end if
    Augmentation:
14:     $\mathcal{S}^i = \mathcal{S}^{i-1} \cup \{j^*\}$
    Residual update:
15:     $\boldsymbol{\alpha}^i = \min_{\boldsymbol{\alpha}} \|\mathbf{D}\boldsymbol{\alpha} - \mathbf{y}\|_2^2$ s.t. $support\{\boldsymbol{\alpha}^i\} = \mathcal{S}^i$
16:     $\mathbf{r}^i = \mathbf{y} - \mathbf{D}\boldsymbol{\alpha}^i$
    Stopping rule:
17:     If a) $i >$ desired iterations, or
        b) $\|\mathbf{r}^i\|_2 < \epsilon_0$, or
        c) $\|\mathbf{r}^i\|_2 > \beta\|\mathbf{r}^{i-1}\|_2$ stop,
        otherwise iterate again.
```

[2] Code available at: http://www.csse.uwa.edu.au/~ajmal/code.html

```
Procedure ForwardSelection
Input: $\mathbf{D} \in \mathbb{R}^{m\times k}$, $\mathbf{y} \in \mathbb{R}^{m\times 1}$, $\mathcal{S}$, $\mathcal{T}$, $f$
Output: $j^*$
1: $\mathcal{R}_0 = \emptyset$
2: for each element $\mathcal{T}_i : i \in \{1,\dots,z\}$ in $\mathcal{T}$ do
3:     $\mathcal{S}^0 = \mathcal{S} \cup \mathcal{T}_i$
4:     $\boldsymbol{\alpha}^0 = (\mathbf{D}_{\mathcal{S}^0}^T\mathbf{D}_{\mathcal{S}^0})^{-1}\mathbf{D}_{\mathcal{S}^0}^T\mathbf{y}$: $\mathbf{D}_{\mathcal{S}^0}$ = Matrix with columns of $\mathbf{D}$ indexed in $\mathcal{S}^0$
5:     $\mathbf{r}^0 = \mathbf{y} - \mathbf{D}_{\mathcal{S}^0}\boldsymbol{\alpha}^0$
6:     for $q = 1$ to $f$ do
7:         Compute $\epsilon(j) = \min_{z_j} \|\mathbf{d}_j z_j - \mathbf{r}^{q-1}\|_2^2$, $\forall j \in \{1,\dots,k\}$, using $z_j^* = \mathbf{d}_\mathbf{j}^\mathbf{T}\mathbf{r}^{q-1}/\|\mathbf{d}_j\|_2^2$
8:         Find the minimizer, $j_0$ of $\epsilon(j) : \forall j \notin \mathcal{S}^{q-1}$, $\epsilon(j_0) \leq \epsilon(j)$
9:         $\mathcal{S}^q = \mathcal{S}^{q-1} \cup \{j_0\}$
10:        $\boldsymbol{\alpha}^q = \min_{\boldsymbol{\alpha}} \|\mathbf{D}\boldsymbol{\alpha} - \mathbf{y}\|_2^2$ s.t. $support\{\boldsymbol{\alpha}\} = \mathcal{S}^q$
11:        $\mathbf{r}^q = \mathbf{y} - \mathbf{D}\boldsymbol{\alpha}^q$
12:    end for
       $\mathcal{R}_i = \mathcal{R}_{i-1} \cup (\sum_{\gamma=0}^{f} \|\mathbf{r}^\gamma\|_2^2)$
13: end for
14: $j^*$ = Element of $\mathcal{T}$ corresponding to the smallest element in $\mathcal{R}_z$.
```

For different iterations of OMP-Star, $|\mathcal{T}|$ is expected to be different, where $|.|$ denotes the cardinality of the set. In a particular iteration, $|\mathcal{T}|$ will be large if $\mathbf{D}$ contains many columns with high correlation to $\mathbf{d}_{j_0}$. On the other hand, if $\mathbf{d}_{j_0}$ has very low correlation with other columns, $|\mathcal{T}| = 1$. The parameter t provides the quantitative boundary for deciding on the high and low correlation. For the cases when $|\mathcal{T}| = 1$, j^* simply points to $\mathbf{d}_{j_0}$ (lines "9" and "10" in Algorithm 2). Otherwise, the algorithm selects the index j^* by calling out to a procedure $ForwardSelection$ (line "12" in Algorithm 2). This procedure works as follows. It picks an element of $\mathcal{T}$ and temporarily augments the selected subspace with the corresponding column of $\mathbf{D}$. It then performs f OMP-like iterations, using this augmented subspace as the selected subspace. Each of these iterations may add a vector (i.e., a column of $\mathbf{D}$) to the temporary subspace. Before the first iteration and after each iteration, the procedure notes the l_2-norm of the residual vector. Once f iterations have been performed, it sums up the $f+1$ values of the noted residuals and saves the cumulative residual. The last $f+1$ columns of the temporarily augmented subspace are then removed. The aforementioned process is repeated for each element of $\mathcal{T}$. Once finished, the procedure finds the minimizer over the cumulative residuals. The element of $\mathcal{T}$ corresponding to this minimizer is considered j^*.

Note that the column corresponding to j^* is a suitable choice in the current main iteration of OMP-Star because it is one of the columns that best correlate with the residual. More importantly, among all the suitable choices, it is *potentially* the quickest in terms of reducing the residual. The later property is ensured by the $ForwardSelection$ procedure that identifies the column by looking forward into f OMP-like iterations. This futuristic identification helps in mitigating the effects of the local optimality issue. This notion is similar to the main idea of A-Star, as mentioned in Section IV. However, there is a discernible difference between the main objectives of the two approaches. Whereas the proposed approach only aims at improving the robustness against the local optima in the solution space, A-Star aims at always finding the globally optimal solution, usually at the cost of very high computation. A-Star uses a cost function that employs an admissible heuristic to ensure the global optimality of the solution. This solution is found by iteratively expanding the current best node of the search tree it parses. Therefore, A-Star keeps cost records of all the expanded nodes and backtracks when a previously expanded node has a cost lower than the currently expanded best node. This makes A-Star an optimal search algorithm; however, its search strategy is computationally very expensive [44].

The A$*$OMP proposed by Karahanoglu and Erdogan [51] directly incorporates the idea of A-Star search into the OMP algorithm. As mentioned in [51], similar to A-Star, the objective of A$*$OMP is to find the optimal solution without particularly considering the computational complexity. Therefore, the computation time of A$*$OMP is generally very high. Compromise over the computational complexity makes A$*$OMP less appealing for the problem of hyperspectral unmixing. In order to maintain the computational advantages of the greedy pursuit strategy, OMP-Star does not aim at the global optimality of the solution. However, it is able to mitigate the effects of local optimality by looking into its future iterations. Since A$*$OMP directly uses the A-Star search strategy, it also requires a cost function to decide on the best node to expand. Generally, combining this cost function with OMP is not straightforward [51]. An appropriately defined cost function also needs to compensate for the different path lengths during the search process, which adds to the complexity of this function. A$*$OMP uses a path pruning strategy to make the search process tractable. Since the proposed algorithm (i.e., OMP-Star) does not directly integrate the A-Star search strategy with OMP, it does not suffer from these limitations. In comparison to A$*$OMP, the proposed algorithm is simpler and computationally much more efficient.

B. OMP-Star+

Reflectances of materials are nonnegative values. In other words, for the problem of sparse unmixing, the columns of $\mathbf{D}$ will only have nonnegative coefficients. Furthermore, $(P_0^{\eta+})$ in (5) dictates that the coefficients of $\boldsymbol{\alpha}$ should always be nonnegative. We make use of these constraints and further tailor OMP-Star for sparse unmixing. We denote this nonnegative version of OMP-Star as OMP-Star+. For conciseness, we only present the changes in OMP-Star that would convert it to OMP-Star+.

The first change must be made in line "5" of Algorithm 2. In the ith main iteration, instead of simply minimizing $\|\mathbf{d}_j z_j - \mathbf{r}^{i-1}\|_2^2$ over z_j, we must also make sure that the minimizers are nonnegative quantities. Mathematically, $\forall j \in \{1,\dots,k\}$

$$\epsilon(j) = \min_{z_j \geq 0} \|\mathbf{d}_j z_j - \mathbf{r}^{i-1}\|_2^2$$
$$= \|\mathbf{r}^{i-1}\|_2^2 - \frac{\max\{\mathbf{d}_j^T\mathbf{r}^{i-1}, 0\}^2}{\|\mathbf{d}_j\|_2^2}.$$

Computing $\epsilon(j)$ as aforementioned implies that all the vectors corresponding to the set $\mathcal{T}$ (line "8" in Algorithm 2) will be almost parallel to the residual vector $\mathbf{r}^{i-1}$. This is different from the case of OMP-Star, where we also allow the vectors to be almost antiparallel to $\mathbf{r}^{i-1}$. Since OMP-Star can use negative elements in $\boldsymbol{\alpha}$, it can later invert the direction of the suitable vectors. However, in OMP-Star+, it is not possible because here we constrain the elements of $\boldsymbol{\alpha}$ to be positive. This constraint is imposed by making a second change in Algorithm 2. We change the optimization problem in line "15" of the algorithm with the following constrained optimization problem:

$$\boldsymbol{\alpha}^i = \min_{\boldsymbol{\alpha}} \|\mathbf{D}\boldsymbol{\alpha} - \mathbf{y}\|_2^2, \quad \text{s.t. } support\{\boldsymbol{\alpha}^i\} = \mathcal{S}^i, \quad \boldsymbol{\alpha} \geq 0.$$

The aforementioned changes impose nonnegativity in the main iteration of OMP-Star. We also need to impose this constraint in the $ForwardSelection$ procedure, in order to fully convert OMP-Star to OMP-Star+. Thus, for OMP-Star+, we compute $\epsilon(j)$ (line "7" of the Procedure) using the following equation:

$$\epsilon(j) = \|\mathbf{r}^{q-1}\|_2^2 - \frac{\max\left\{\mathbf{d}_j^T\mathbf{r}^{q-1}, 0\right\}^2}{\|\mathbf{d}_j\|_2^2}, \forall j \in \{1, \ldots, k\}.$$

Similarly, the optimization problem in line "10" of the Procedure is replaced with the corresponding constrained optimization problem. These changes ensure that the selection of j^* made by the procedure follow the nonnegativity constraint.

V. Coherence Reduction

In SA problems, coherence μ is one of the most fundamental characteristics associated with $\mathbf{D}$ [45]. The coherence of a matrix denotes the maximum absolute inner product between any two distinct columns. That is

$$\mu = \max_{i,j;j\neq i} \frac{\left|\mathbf{d}_i^T\mathbf{d}_j\right|}{\|\mathbf{d}_i\|_2\|\mathbf{d}_j\|_2}. \tag{13}$$

Consider the following noiseless SA problem:

$$(P_0): \quad \min_{\alpha} \|\boldsymbol{\alpha}\|_0 \quad \text{s.t.} \quad \mathbf{D}\boldsymbol{\alpha} = \mathbf{y}. \tag{14}$$

If $\mathbf{y}$ is a linear combination of p distinct columns of $\mathbf{D}$, then we are interested in the upper bound on p, such that the support of the solution $\boldsymbol{\alpha}$ *always* points to the correct p columns of $\mathbf{D}$. Tropp [45] showed that this bound depends on μ, as follows:

$$p < \frac{1}{2}\left(1 + \frac{1}{\mu}\right). \tag{15}$$

The bound has been proven in [45] for OMP and BP. This bound is also valid for the algorithms proposed in this paper.[3] Elad *et al.* [31] modified the bound by replacing μ in (15) with $\mu_\triangle$, where $\mu_\triangle$ is the coherence of $\mathbf{PD}$. Here, $\mathbf{P}$ must be a square invertible matrix. This modification is achieved by replacing the constraint in (P_0) by $\mathbf{PD}\boldsymbol{\alpha} = \mathbf{Py}$.

Note that the bound in (15) is pessimistic because μ is the worst case measure. If the columns of $\mathbf{D}$ have high correlation, the inequality (15) becomes practically meaningless. For instance, according to the aforementioned inequality, two mixed signals in $\mathbf{y}$ (i.e., $p = 2$) requires $\mu < 0.33$ for their guaranteed exact identification. However, $\mu \approx 1$ when columns of $\mathbf{D}$ have high correlation, as in sparse unmixing [11]. Therefore, the inequality (15) is not meaningful for sparse unmixing. However, sparse unmixing can benefit from two important observations based on the aforementioned discussion. 1) Low correlation among the columns of $\mathbf{D}$ favors accurate support of the solution. 2) An invertible matrix $\mathbf{P}$ can modify this correlation without affecting the results. Based on these observations, we focus on finding the invertible matrix $\mathbf{P}$, which reduces the correlation among the columns of $\mathbf{D}$, thereby improving the accuracy of the solution support.

Tsai and Philpot [46] used derivatives to analyze remote sensing data with high spectral resolution. Bieniarz *et al.* [49] recently showed that, generally, the correlation among spectra can be reduced with the help of derivatives. Therefore, we use the same tool to construct the desired matrix $\mathbf{P}$ for the hyperspectral unmixing problem. For a spectra captured at constant wavelength intervals, we can formulate the derivative operator ($\triangle$) in its generic form as follows:

$$\triangle_s^o d_b = \frac{1}{(\Delta\lambda)^o}\sum_{i=0}^{o}(-1)^i\binom{o}{i}d_{b+[(o-i)\times s]} \tag{16}$$

where d_b is the reflectance of the spectra at band b. Among the parameters of the operator, o is the *order of the derivative*, and s is the *band difference*. The band difference denotes the number of bands over which the slope is calculated. Domains of both of the aforementioned parameters are positive integers. On the right-hand side, $\Delta\lambda$ denotes the *wavelength difference* between the bands over which the slope is computed.

We exemplify the construction of $\mathbf{P}$ from (16) for particular values of s and o. Consider a spectra given as a vector $\mathbf{d} \in \mathbb{R}^m$, for which the wavelength difference between two consecutive samples is 0.01 μm. If we choose $o = 1$ and $s = 2$ in (16), the equation simplifies to

$$\triangle_2^1 d_b = \frac{d_{b+2} - d_b}{2 \times 10^{-9}}. \tag{17}$$

By definition, the aforementioned derivative can be computed for each band of $\mathbf{d}$, except the last $s = 2$ bands. For those bands, the derivative operation keeps the original reflectance values of the signal. $\triangle_2^1$ operates band by band. It is also possible to take the derivative of the complete vector $\mathbf{d}$ by left multiplying it with $\gamma\mathbf{P}$, where $\gamma \in \mathbb{R}$ is a constant, and $\mathbf{P}$ is a matrix in $\mathbb{R}^{m\times m}$. For $\triangle_2^1$, $\gamma = -1/(2 \times 10^{-9})$ and $\mathbf{P}$ is a matrix with its elements $p_{i,i} = 1, \forall i \in \{1, \ldots, m\}$ and $p_{i,i+2} = -1, \forall i \in \{1, \ldots, m-2\}$. All other elements of $\mathbf{P}$ are zero.

[3] The futuristic greedy heuristic will give the same result as the greedy heuristic, if the latter is guaranteed to find the optimal result. This is exactly the case when the data satisfy the inequality (15).

The square matrix $\mathbf{P}$ created for $\triangle_2^1$ is an invertible matrix because it has 1s at its diagonal and nonzero elements only in the upper triangle. In fact, this property holds for all the $\mathbf{P}$ matrices created with the aforementioned strategy for any values of s and o satisfying the inequality[4] $(1 + os) \leq m$. The $\mathbf{P}$ matrix thus created can directly be left multiplied with $\mathbf{D}$ to take derivatives of all the spectra in the library simultaneously. We can either choose to ignore γ or multiply it with $\mathbf{P}$ and use this modified matrix. We prefer the latter choice as it corresponds to the exact definition of derivatives used in [46]. In general, the derivative operator is able to enhance the dissimilarities between two seemingly identical spectra. This results in the reduction of correlation between the spectra.

VI. Experiments With Synthetic Data

Here, we evaluate the proposed approach with synthetic data. The experiments performed on synthetic data are significant because the endmembers and their corresponding fractional abundances are known, enabling quantitative analysis of the unmixing. After introducing the data used in the experiments, we first discuss the preprocessing of the data with derivatives. This is followed by the experiments for evaluating the proposed algorithms for endmember identification and fractional abundance estimation.

A. Data

In the experiments with synthetic data, we created our matrix $\mathbf{D}$ from the NASA Jet Propulsion Laboratory's Advanced Space-borne Thermal Emission and Reflectance Radiometer (ASTER) Library.[5] This library contains pure spectra of 2400 natural and man-made materials. For the experiments, we chose 425 of these spectra. The spectra were chosen such that their coherence μ is 0.9986. We kept $\mu < 1$ in order to ensure that all the selected spectra were unique. We resampled the spectra in the wavelength range from 0.4 to 2.5 μm, at a constant interval of 10 nm. This resulted in 224 samples per spectra. We performed this resampling, in order to match the sampling strategy of NASA's Airborne Visible Infrared Imaging Spectrometer (AVIRIS) [29]. From the resampled spectra, we dropped the first 24 bands because of zero or very low reflectance values. We created $\mathbf{D}$ with the processed spectra, which made it a 200×425 matrix. Henceforth, we use the term "library" for $\mathbf{D}$. All the techniques compared in the experiments use the same library. Among the 425 spectra in the library, 285 belong to minerals, 15 belong to soils, and 125 represent rocks.

For the unmixing problem, we create a mixed signal/pixel $\mathbf{y}$ by randomly selecting p spectra from the library. We simulate the hyperspectral data cube as 500 mixed pixels, in which each pixel comprises p *different* randomly selected spectra/endmembers. Thus, we simulate a highly mixed scenario. Pixel-based algorithms enjoy good performance in these scenarios because they do not depend on the contextual or spatial information in the data. Since the proposed algorithms are pixel based, we compare their performances only with the other pixel-based algorithms. Following the experimental protocol in [11], we draw the fractional abundances of the endmembers in each pixel of the hyperspectral data cube from a Dirichlet distribution. Therefore, the fractional abundances in each pixel sum up to 1, and the ASC holds for each pixel. Unless stated otherwise, the results reported as follows are the *mean* values computed over the hyperspectral data cube.

B. Analysis of Derivatives

To analyze the correlation reduction characteristic of derivatives, we introduce the *mean coherence* $(\bar{\mu})$ for a spectral library $\mathbf{D}$

$$\bar{\mu} = \frac{1}{k}\left(\sum_{i=1; i\neq j}^{k} \max_j \frac{\left|\mathbf{d}_i^T \mathbf{d}_j\right|}{\|\mathbf{d}_i\|_2 \|\mathbf{d}_j\|_2}\right) \tag{18}$$

which maximizes the correlation of each spectra $\mathbf{d}_i$ over all the other spectra in the library and takes the mean of the computed maximum values. Note that spectral derivatives *generally* reduce the correlation among the spectra. Since μ is only the worst case measure, it is not possible to correctly analyze this property of the spectral derivatives using μ. Compared to μ, $\bar{\mu}$ gives a better picture of the similarity among the spectra in the library because every spectra in $\mathbf{D}$ contributes to $\bar{\mu}$. The mean coherence of $\mathbf{D}$ in our experiments is 0.9974. We calculated the difference between the mean coherences of $\mathbf{D}$ before taking its derivative (i.e., $\bar{\mu}$) and after taking the derivative (i.e., $\bar{\mu}_\triangle$). The greater is this value, the more advantageous is the derivative operation for the sparse unmixing problem. Fig. 2(a) shows the plots for $\bar{\mu} - \bar{\mu}_\triangle$ (scaled by a factor of 100) as a function of band difference s. The plots are shown for the first-, second-, and third-order derivatives.

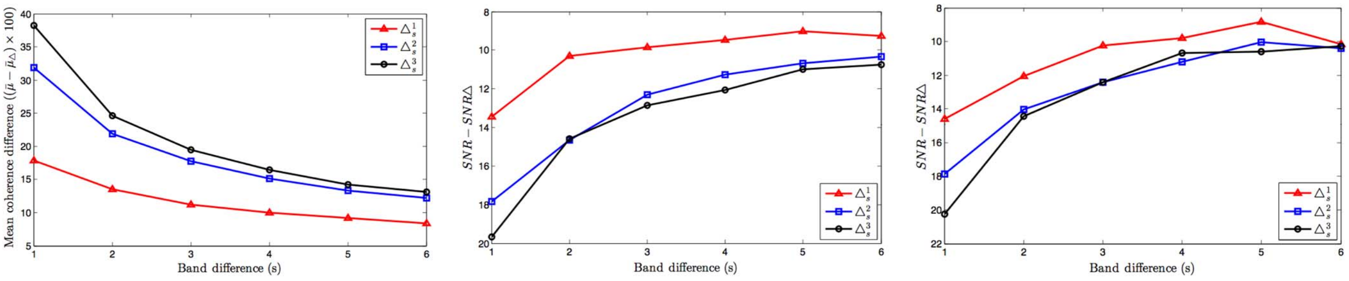
In order to benefit from derivatives in sparse unmixing, they must also be operated on the mixed pixels. The mixed pixels can be noisy, whereas a derivative operation is usually sensitive to noise. This is because derivatives amplify the sudden changes in the reflectance patterns of spectra. Keeping in view the noise sensitivity of derivatives, we analyzed the effects of noise amplification caused by derivatives. For this purpose, we corrupted each spectra in $\mathbf{D}$ with additive noise to create a library $\tilde{\mathbf{D}}$. Each spectra in $\tilde{\mathbf{D}}$ has SNR = 35 dB. We took the derivatives of $\tilde{\mathbf{D}}$ with different values for the parameters s and o. Each time, we computed $\text{SNR}_\triangle$ using the following formula:

$$\text{SNR}_\triangle = 10\log_{10}\left(\frac{E\left[\|\mathbf{d}_\triangle\|_2^2\right]}{E\left[\|\mathbf{d}_\triangle - \tilde{\mathbf{d}}_\triangle\|_2^2\right]}\right). \tag{19}$$

The aforementioned expression estimates the SNR of the differentiated library $\tilde{\mathbf{D}}_\triangle$ in decibel. In this expression, $\mathbf{d}_\triangle$ denotes the spectra in the differentiated library $\mathbf{D}_\triangle$, $\tilde{\mathbf{d}}_\triangle$ denotes the spectra in $\tilde{\mathbf{D}}_\triangle$, and $E[.]$ is the expectation operator. In our experiments, we have used white noise and correlated noise. Therefore, we separately analyze their amplification with derivatives. We have followed Bioucas-Dias and Nascimento

[4] The inequality follows from the general definition of the derivative given in (16) and gives the upper bound on the parameter values.

[5] Available at http://speclib.jpl.nasa.gov



(a) (b) (c)

Fig. 2. Analysis of derivatives: (a) Difference between the mean coherence of the library before and after taking the derivative (scaled by a factor of 100), as a function of the band difference s. (b) Difference between SNR and SNR$_\triangle$, as a function of s for white noise. (c) Difference between SNR and SNR$_\triangle$, as a function of s for correlated noise. SNR $= 35$ dB in (b) and (c).

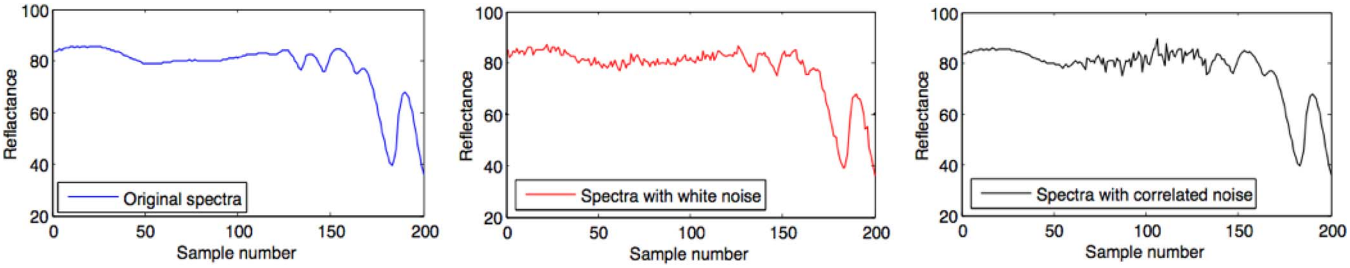


Fig. 3. Example of white and correlated noise: In the given original reflectance spectra of a mineral, white and correlated noise have been added to obtain SNR $= 35$ dB. The white noise is generated by choosing $\eta = 500$, whereas $\eta = 18$ for the correlated noise. Band 50 to 150 are corrupted with the correlated noise, with maximum noise power around band 100.

[52] to generate both types of the additive noise. Let $\mathbf{R}_n = \text{diag}(\sigma_1^2, \ldots, \sigma_m^2)$ be the correlation matrix of the additive noise, where the diagonal of the matrix follows a Gaussian shape centered at the band $m/2$ and

$$\sigma_i^2 = \frac{\sigma^2}{\eta\sqrt{2\pi}} e^{\frac{-\left(i-\frac{m}{2}\right)^2}{2\eta^2}} \qquad \forall i \in \{1,\ldots,m\}. \tag{20}$$

In (20), the parameter η controls the variance of the Gaussian shape (i.e., $\eta \to 0$ generates a single-band noise, and $\eta \to \infty$ results in white noise), whereas

$$\sigma^2 = \frac{\sum_{j=1}^{k} \|\mathbf{d_j}\|_2^2}{k \times 10^{\left(\frac{\text{SNR}}{10}\right)}} \tag{21}$$

controls the total noise power. Further details on the aforementioned noise model can be found in [52]. For the fixed value of SNR $= 35$ dB, we vary the value of η to obtain the desired type of noise. In all of our experiments, we have used $\eta = 500$ to generate the white noise and $\eta = 18$ to generate the correlated noise. Fig. 3 shows the results of using these values in generating the additive noise.

Fig. 2(b) and (c) plot the difference in SNR and SNR$_\triangle$, as a function of the band difference s for different values of o, for white and correlated noise, respectively. The plots in the figures remain qualitatively similar for different values of SNR (in the range from 20 to 100 dB). We restricted the analysis only up to third-order derivatives in Fig. 2 because the noise amplification for higher order derivatives becomes too large, rendering the derivative operation unbeneficial for sparse unmixing.

Based on the plots in Fig. 2, we selected $s = 5$ and $o = 1$ for the derivative operation in our experiments, which are presented as follows. These values of the parameters provide a reasonable tradeoff between the correlation reduction and noise amplification at SNR $= 35$ dB. Notice that the effect of noise amplification reduces with the increase in the value of s. This is due to the smoothing caused by the derivative operation, which is illustrated in Fig. 4. In general, for small values of SNR, it is more beneficial to use larger values of s to control the noise amplification, whereas for high SNR of the spectral signatures, s should be kept small (e.g., 1 or 2) to fully exploit the coherence reduction ability of derivatives.

C. Experiments for the Endmember Identification

1) Unmixing Fidelity: A hyperspectral unmixing approach should be evaluated according to the two goals of unmixing mentioned in Section II-B. For the first goal of "finding the correct set of endmembers," we introduce the evaluation metric of *unmixing fidelity* of a sparse unmixing solution, i.e., $\Phi(\boldsymbol{\alpha}) \to [0, 1]$. If $\mathcal{P} = \{x | x$ is the index of an endmember in $\mathbf{D}\}$ and $\mathcal{A} = \{a | a$ is a nonzero elements in $\boldsymbol{\alpha}\}$, then

$$\Phi(\boldsymbol{\alpha}) = \frac{|\mathcal{P} \bigcap \mathcal{A}|}{|\mathcal{A}|}. \tag{22}$$

Here, $|.|$ denotes the cardinality of the set. Unmixing fidelity calculates the fraction of the correct support in the solution $\boldsymbol{\alpha}$. It does not take into account the fractional abundances because, with known correct support of $\boldsymbol{\alpha}$, it is always possible to compute the fractional abundances separately (e.g., with the least squares method).

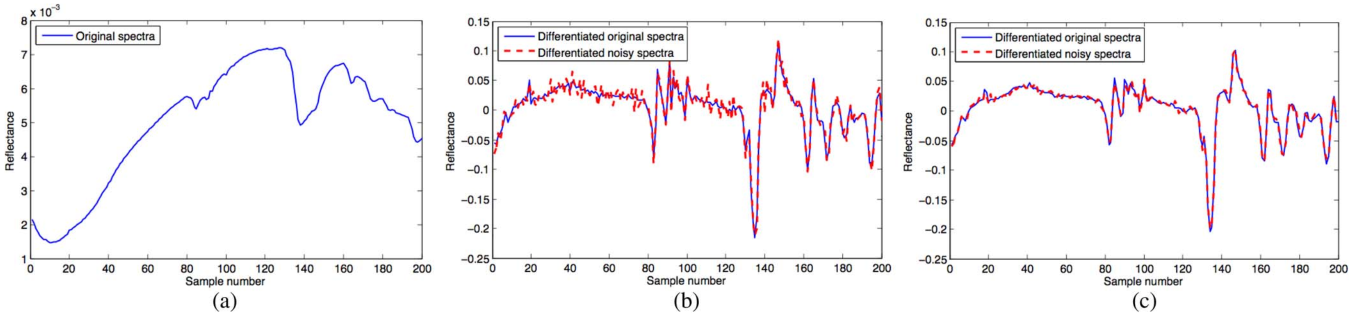


Fig. 4. Smoothing effect caused by increasing s in derivatives: (a) Spectra of a mineral. (b) Solid line: derivative of the pure spectra. Dotted line: derivative of the noisy spectra. Derivatives taken with $o = 1$ and $s = 1$. (c) Derivatives taken with $o = 1$ and $s = 2$. The resulting plots are smoother, and the noise amplification has reduced.

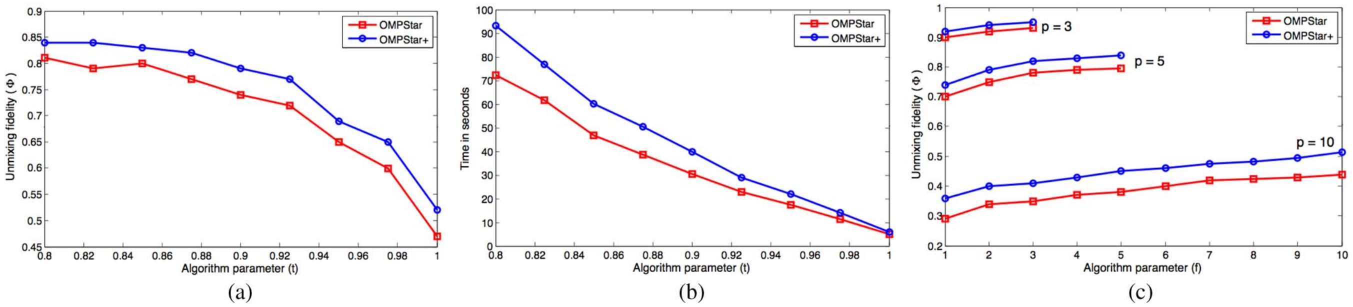


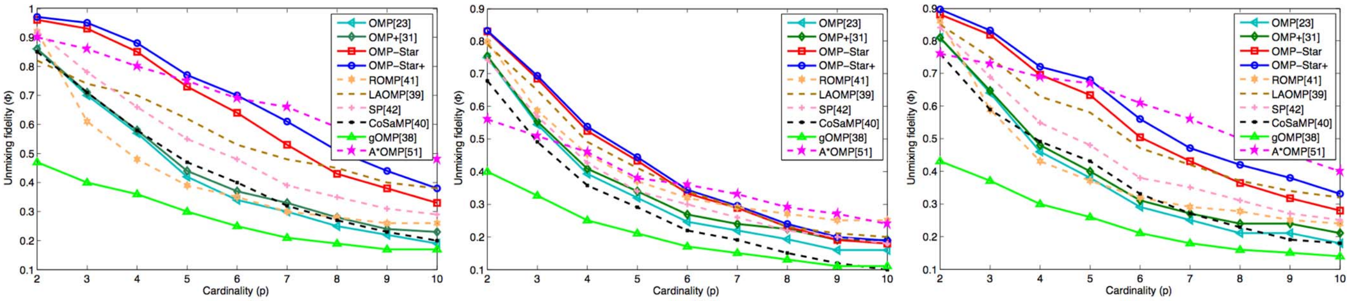
Fig. 5. Parameter selection with noiseless training data: The plots have been obtained for $p = 5$. (a) Unmixing fidelity as a function of t, with $f = 2$. (b) Computation time (in seconds) as a function t, with $f = 2$. (c) Unmixing fidelity as a function of f, with $t = 0.92$. For each group of the curves, the value of p is mentioned beside it.

2) Parameter Value Selection: Before experimenting with the proposed algorithms, we must select the values of the parameters t and f for the algorithms. In an iteration, t decides the columns of $\mathbf{D}$ to be chosen for the $ForwardSelection$ procedure, whereas f is the number of OMP-like iterations performed by the procedure to select the best column. We selected the values of these parameters by experimenting on separate training data, which were created with the same protocol as mentioned in Section VI-A. In these experiments, the algorithms were given the library and the pixel after operating $\triangle_5^1$ on them. For the library, the derivative was operated after normalizing its columns in l_1-norm. Note that we follow the same procedure of preprocessing the data for all the greedy algorithms in the experiments to follow. For the selection of values of parameters, we chose the data to be noise free, with $p = 5$ for each pixel. We analyzed the unmixing results for different values of the parameters for the proposed algorithms. These parameters offer a tradeoff between Φ and the computation time of the algorithms. Fig. 5 shows plots of the results that also offer insights into the general behavior of the algorithms for different values of the parameters.

Fig. 5(a) shows that larger values of t reduce the unmixing fidelity of the solution. However, they also decrease the computation time of the algorithms in a linear fashion, as shown in Fig. 5(b). This trend is the same for OMP-Star and OMP-Star+. Fig. 5(c) shows the effects of increasing f on Φ. In the figure, the three groups of curves correspond to three different values of p. It is clear from the figure that increasing f generally improves the unmixing fidelity of the solution. This statement is valid for all the values of p. Notice that, with an increase in p, the unmixing fidelity decreases. This is a general trend for sparse unmixing algorithms. However, the fact that the cardinality of a mixed pixel in practical scenarios is usually on the order of four to five [11] mitigates the adverse effects of this trend. For the sake of thoroughness of the analysis, we used values of p as high as 10 in our experiments.

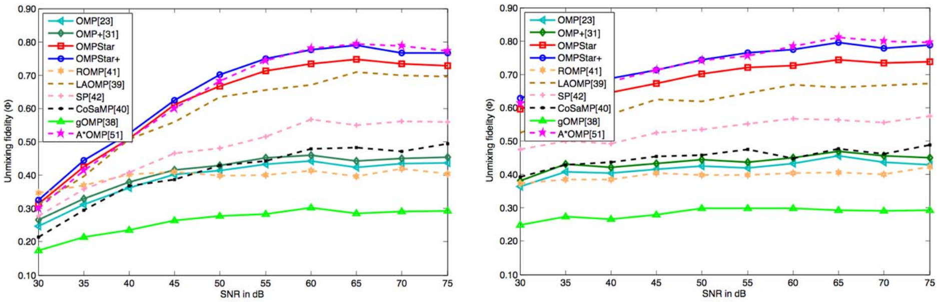
3) Results: Based on the aforementioned experiments, we selected $t = 0.92$ and $f = 2$ for OMP-Star and OMP-Star+. We have performed experiments for the comparison of these algorithms with all the other greedy pursuit algorithms mentioned in Section III. Fig. 6 shows this comparison in terms of unmixing fidelity of the solutions as a function of p. The experiments were performed for the cases when the mixed pixels were noise free [see Fig. 6(a)] and noisy. For the noisy case, we chose SNR $= 35$ dB with additive white noise [see Fig. 6(b)] and correlated noise [see Fig. 6(c)].

In Fig. 6, the algorithms represented by dotted lines are impractical choices for hyperspectral unmixing; however, we have included their results for the sake of thoroughness of the analysis. Among these algorithms, SP, CoSaMP, and ROMP *must* know the value of p *a priori*. In hyperspectral unmixing, this value is usually unknown. We have also shown LAOMP and A$*$OMP with dotted lines because their computational time is so high that they loose the computational advantages of the greedy approach (see Section VIII). A$*$OMP is one algorithm that is consistently able to show better results than the proposed algorithm for $p \geq 5$. However, the proposed algorithms have computed these results nearly 200 times faster than A$*$OMP. It is also worth mentioning here that p is usually 4 to 5 in practical scenarios [11].



(a) (b) (c)

Fig. 6. Performance comparison of greedy algorithms: The plots show unmixing fidelity of the solutions found by different algorithms as a function of cardinality of the mixed pixels. (a) Noiseless case. (b) 35-dB SNR with white noise. (c) 35-dB SNR with correlated noise.



(a) (b)

Fig. 7. Unmixing fidelity as a function of SNR: For all the plots, $p = 5$. (a) White noise. (b) Correlated noise.

For the experiments, the data have been preprocessed with spectral derivatives for all the greedy algorithms, including the proposed ones. As shown in Fig. 2(a), the preprocessing reduces the coherence between the spectral signatures. However, this reduction is not sufficient to achieve accurate endmember identification using the greedy algorithms. Therefore, even after the preprocessing of the data, a greedy algorithm must show robustness to the high coherence of the data for better results. In the proposed algorithms, this robustness is achieved by the futuristic heuristic, which resulted in the better performance.

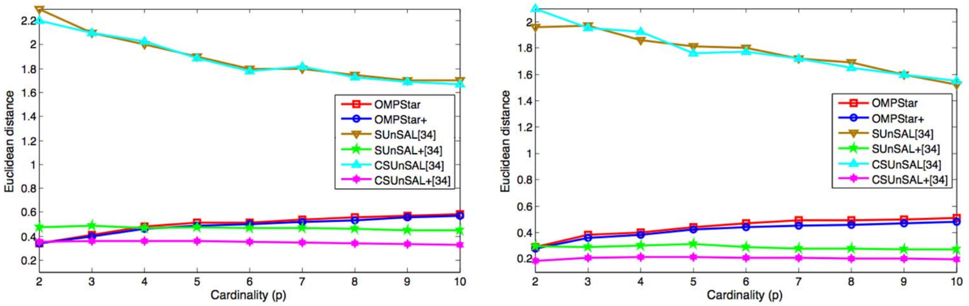
Here, we have assumed that p is known beforehand (the assumption will be removed in the next section). Therefore, the proposed algorithms use stopping rule (a) in line "17" of Algorithm 2 (see Section IV). All the other algorithms also use the same stopping rule in this analysis. For LAOMP, we have selected $L = 3$, and for gOMP, we have chosen $L = 2$. Correct values of p have been given to OMP, OMP+, ROMP, SP, CoSaMP, LAOMP, and gOMP as their input parameters. For A$*$OMP, we have chosen $I = 3$, $B = 40$, $\beta = 1.2$, $Kmax = 10$, and $\epsilon = 10^{-7}$. These parameters respectively represent the "number of initial search paths," "the branching factor in each iteration," "the parameter of adaptive auxiliary function," "maximum allowed sparsity level," and "the residual power for termination of the algorithm." We have chosen the "adaptive" cost function for A$*$OMP (see [51] for details) in all of our experiments, as this function resulted in the best performance of the algorithm. For each of the algorithms, we have optimized the parameter values using separate training data. With the aforementioned parameter settings, we have also analyzed the algorithms for the unmixing fidelity as a function of SNR of the mixed signals, with $p = 5$. Results of these experiments are shown in Fig. 7. It is shown in the figure that the proposed algorithms are able to consistently perform well.

D. Experiments for Fractional Abundance Estimation

Once the endmembers have been identified, the unmixing problem reduces to computing the fractional abundances of the identified endmembers. If $\boldsymbol{\alpha} \in \mathbb{R}^k$ is the estimated fractional abundance vector and $\boldsymbol{\alpha}_o \in \mathbb{R}^k$ is the vector containing the actual fractional abundances, then the Euclidean distance $\|\boldsymbol{\alpha}_o - \boldsymbol{\alpha}\|_2$ is a suitable metric for evaluating the solution for fractional abundance estimation. Fig. 8 uses this metric to compare the performance of the proposed algorithms against the popular unmixing algorithms, which are based on convex relaxation of the problem. Among these, SUnSAL+ and CSUnSAL+ are the nonnegative variants of SUnSAL and CSUnSAL. Fig. 8 shows that the performance of the proposed algorithms is comparable to these well-known algorithms. We should mention that the proposed algorithms achieve these results much faster than the other algorithms shown in the figure. Comparison of execution times of all the algorithms is given in Section VIII.

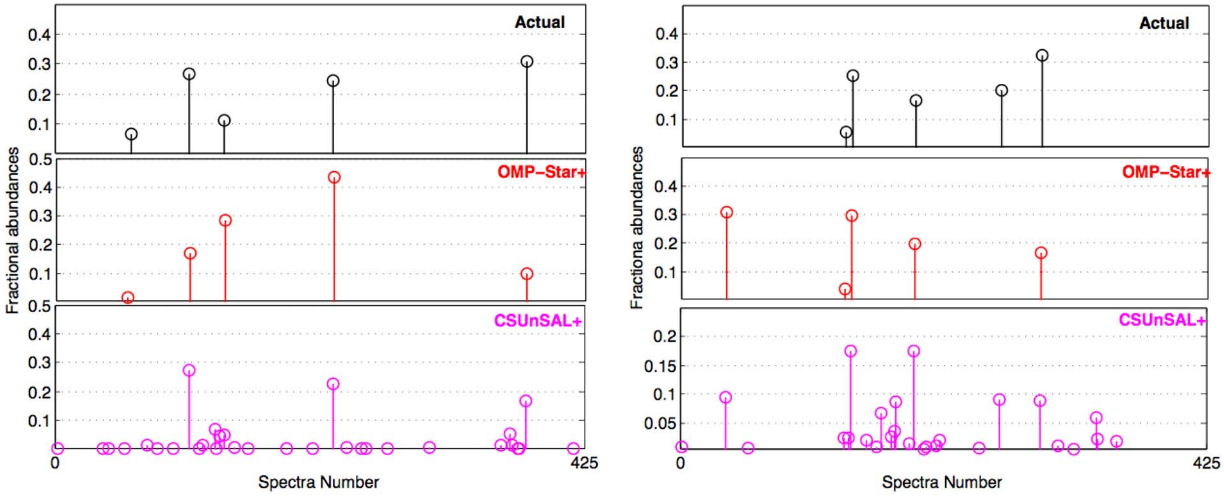
For the proposed algorithms, the results shown in Fig. 8 are obtained using the following strategy:

1) Compute $\boldsymbol{\alpha}$ after taking the derivative of the data, as mentioned in Section VI-C2.
2) Create $\mathbf{D}_{\mathcal{A}}$ from the columns of $\mathbf{D}$ corresponding to the support of $\boldsymbol{\alpha}$.
3) Use $\mathbf{D}_{\mathcal{A}}$ and $\mathbf{y}$ as inputs for CSUnSAL+, which calculates the fractional abundances.
4) Replace the fractional abundances in $\boldsymbol{\alpha}$ with the corresponding new values.



(a) (b)

Fig. 8. Euclidean distance between the original fractional abundance vector and the computed fractional abundances vector: (a) 35-dB SNR with white noise. (b) 35-dB SNR with correlated noise.



(a) (b)

Fig. 9. Examples of fractional abundances estimated by OMP-Star+ and CSUnSAL+: The horizontal axis shows the indexes of spectra in $\mathbf{D}$. (First row) Actual fractional abundances of the five endmembers. (Second row) Fractional abundances of the endmembers computed by OMP-Star+. (Third row) Fractional abundances of the endmembers computed by CSUnSAL+. $\|\boldsymbol{\alpha}_0 - \boldsymbol{\alpha}\|_2$ for OMP-STAR+ and CSUnSAL+ are 0.35 and 0.20 in (a) and 0.42 and 0.30 in (b).

The aforementioned strategy is similar to the library pruning concept in [32], but we use the algorithm CSUnSAL+, only for estimating the fractional abundances. In Fig. 8, the value of the regularization parameter λ (see (7) in Section II-B) is set to 10^{-5} for all of the convex relaxation-based algorithms, whereas the tolerance for the primal and dual residuals (see [34] for details) has been set to 10^{-3} for SUnSAL and CSUnSAL, and 10^{-5} for SUnSAL+ and CSUnSAL+. All of these values have been optimized on separate training data. We have used the same parameter settings for CSUnSAL+, when it is used in the aforementioned strategy. For the convex relaxation algorithms, we have reported the results on the original data. When differentiated data are used with these algorithms, the amplified noise causes problems in accurate estimation of abundances. In the presence of high noise, setting the algorithms for high-fidelity signal reconstruction leads to poor estimates of abundances because the computed abundances also approximate the dominant noise component. To reduce this effect, we can set the algorithms with high tolerance for signal reconstruction. However, it does not solve the problem completely. Due to high tolerance, the underlying original signal is also poorly approximated by the estimated abundances, resulting in compromised abundance estimation. Furthermore, increasing the tolerance beyond a certain range drastically aggravates the accuracy of the computed abundances. Therefore, choosing the right tolerance value becomes nontrivial. We avoid this issue in the case of the greedy algorithms because of the aforementioned strategy. Since the abundances are finally computed on the original data (and not on the differentiated data) in step 4, we do not encounter the problems caused by the noise amplification. We have also experimented with the convex relaxation-based algorithms using the differentiated data. However, despite careful optimization of the parameters, the abundances estimated by these algorithms using the differentiated data were less accurate, as compared to those computed with the original data. Therefore, we have reported the results on the original data in Fig. 8. Notice that, we have also not included the convex relaxation-based algorithms in the comparison for unmixing fidelity in Section VI-C. This is because their results are not comparable with the proposed algorithms for that metric. For instance, Fig. 9 shows two examples of the solutions found by OMP-Star+ and CSUnSAL+. In both cases, for the five actual endmembers (first row), OMP-Star+ identifies five endmembers (second row), out of which four are correctly identified. This results in $\Phi(\boldsymbol{\alpha}) = 0.8$ in both cases. On the other hand, CSUnSAL+ identifies several endmembers with nonzero fractional abundances. In

TABLE I
COMPARISON OF GREEDY ALGORITHMS FOR FRACTIONAL ABUNDANCE ESTIMATION: BELOW EACH ALGORITHM'S NAME, THE COLUMN SHOWS THE VALUES OF $\|\boldsymbol{\alpha}_o - \boldsymbol{\alpha}\|_2$ FOR THE CARDINALITY OF THE MIXED PIXELS IN THE FIRST COLUMN. SNR $= 35$ dB FOR THE MIXED SIGNALS WITH WHITE NOISE

Cardinality	OMP	OMP+	OMP-Star	OMP-Star+	ROMP	LAOMP	SP	CoSaMP	gOMP	A*OMP
2	0.413	0.408	0.311	0.305	0.359	0.351	0.393	0.476	0.359	0.501
3	0.508	0.498	0.402	0.392	0.511	0.438	0.517	0.586	0.421	0.493
4	0.564	0.552	0.461	0.456	0.560	0.485	0.560	0.632	0.470	0.485
5	0.580	0.552	0.521	0.504	0.573	0.480	0.582	0.661	0.496	0.480
6	0.573	0.561	0.530	0.518	0.556	0.512	0.584	0.671	0.488	0.467
7	0.577	0.541	0.545	0.530	0.545	0.507	0.584	0.676	0.490	0.451
8	0.575	0.539	0.561	0.551	0.535	0.522	0.586	0.681	0.488	0.449
9	0.581	0541	0.571	0.562	0.527	0.511	0.581	0.673	0.487	0.453
10	0.591	0.532	0.579	0.569	0.519	0.497	0.586	0.661	0.489	0.459

TABLE II
EXPERIMENTS OF TABLE I, REPEATED WITH CORRELATED NOISE

Cardinality	OMP	OMP+	OMP-Star	OMP-Star+	ROMP	LAOMP	SP	CoSaMP	gOMP	A*OMP
2	0.406	0.405	0.328	0.324	0.355	0.348	0.381	0.448	0.355	0.398
3	0.503	0.495	0.378	0.370	0.539	0.392	0.479	0.501	0.417	0.386
4	0.556	0.541	0.423	0.404	0.582	0.437	0.534	0.589	0.462	0.379
5	0.571	0.544	0.460	0.433	0.582	0.449	0.554	0.604	0.485	0.370
6	0.551	0.521	0.466	0.433	0.559	0.457	0.524	0.598	0.455	0.372
7	0.551	0.517	0.468	0.450	0.547	0.451	0.534	0.610	0.457	0.369
8	0.537	0.498	0.481	0.447	0.527	0.444	0.518	0.602	0.449	0.376
9	0.532	0.488	0.501	0.469	0.526	0.439	0.529	0.609	0.450	0.382
10	0.529	0.486	0.502	0.467	0.507	0.443	0.520	0.613	0.450	0.390

Fig. 9(a), noting that many of the nonzero abundances are very small, we can set a threshold below which the abundances can be neglected. This can improve the unmixing fidelity of the solution. However, choosing the value of the threshold becomes tricky in the cases similar to Fig. 9(b), where some correctly identified endmembers have fractional abundances close to or smaller than the wrongly identified endmembers. In Fig. 9(b), choosing 0.05 as a threshold would result in the detection of eight endmembers, out of which only four would be correct, whereas a value of 0.1 would give only two endmembers.

For the results shown in Figs. 8 and 9, we did not assume prior knowledge of the cardinalities of the mixed pixels. Therefore, the proposed algorithms use stopping rule (c) in line "17" of Algorithm 2, with $\beta = 0.9$. The value of the residual decay parameter β was chosen with the help of separate training data. In order to compare the proposed algorithms with the other greedy algorithms for fractional abundance estimation, we again assumed that the value of p is known for these algorithms (the proposed algorithms do not use the value of p here as well). In this case, for all the greedy algorithms, including the proposed algorithms, we computed the fractional abundances following the aforementioned strategy with a small variation in step 3. Instead of using CSUnSAL+ in this step, we used the nonnegative least squares method to estimate the fractional abundances. The results are shown in Table I for the data with Gaussian white noise and in Table II for the data with correlated noise. To obtain these results, we normalized $\mathbf{D}_{\mathcal{A}}$ and $\mathbf{y}$ in l_1-norm before computing the fractional abundances. According to the analytical results of Theorem 1 in Section II-C, this automatically imposes the ASC along the nonnegativity constraint over the least squares method. Here, we have taken advantage of the results of the theorem only in computing the fractional abundances. Generally, the process of differentiation can introduce some negative values in a nonnegative matrix. However, since $\mathbf{D}_{\mathcal{A}}$ and $\mathbf{y}$ contain the undifferentiated signals, the results of the theorem hold for these data. We emphasize that, in all the experiments in this work, we have used the differentiated data only in identifying the correct set of the endmembers of the mixed pixels using the greedy algorithms (including the proposed algorithms). The fractional abundances are always computed using the original spectral signatures. Therefore, we have been able to take advantage of the results of Theorem 1 for all the greedy algorithms. It is worth mentioning that spectral derivatives have never been exploited for greedy algorithms earlier in this manner.

One interesting observation in these tables is that, for $p \geq 5$, some of the algorithms perform slightly better than the proposed algorithms. This is particularly true for gOMP. However, gOMP's performance for unmixing fidelity is the poorest among the greedy algorithms (see Figs. 6 and 7). The reason for this phenomenon is simple. For the large values of p, the average true fractional abundance of the endmembers becomes very small for a pixel. Algorithms that overestimate the number of endmembers in a pixel (e.g., gOMP) generally assign small values to all the fractional abundances because of ASC. For the correctly identified endmembers, small values become good approximations of the actual fractional abundances. At the same time, small fractional abundances are also good approximations for the falsely identified endmembers (as their true fractional abundances are zero). This results in an overall reduction of $\|\boldsymbol{\alpha}_0 - \boldsymbol{\alpha}\|_2$. The algorithms that do not overestimate the number of endmembers can get heavily penalized when they detect wrong endmembers, when p is high. This is because they end up assigning relatively large values to the fractional abundances when the actual values are zero. Despite this phenomenon, in our opinion, $\|\boldsymbol{\alpha}_0 - \boldsymbol{\alpha}\|_2$ is still a good metric for evaluation of fractional abundance estimations as it accounts for all the endmembers identified by an algorithm. This metric gives a clear picture of the results, particularly for $p \leq 5$ (which corresponds to practical scenarios). The root-mean-square error metric used in [10] and [32] is evaluated with only the correctly identified endmembers. Therefore, it generally favors the overestimating algorithms for all the values of p.

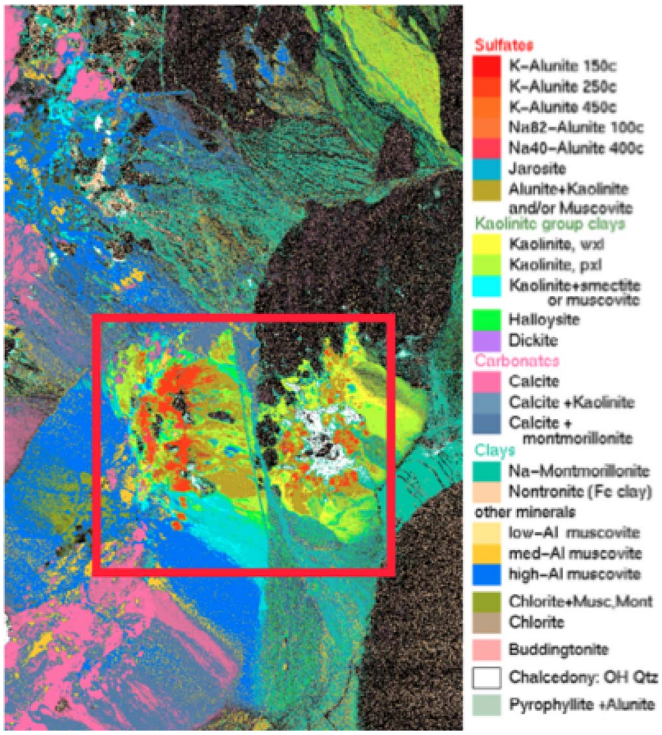


Fig. 10. Mineral map of Cuprite mining district, Nevada: The area inside the red rectangle corresponds to the hyperspectral data cube used in the experiments. The map is made available by USGS at http://speclab.cr.usgs.gov/cuprite95.tgif.2.2um_map.gif.

VII. Experiments With Real Data

Here, we evaluate the proposed algorithms on real-world hyperspectral data. We compute the abundance maps of the materials for the well-known AVIRIS Cuprite data set.[6] This data set was collected over a region in the Cuprite mining district, Nevada. The mineral classification map for this region, made available by the U.S. Geological Survey (USGS), is shown in Fig. 10. The classification map shown in the figure is computed with the Tricorder software product.[7] In our experiments, we use the hyperspectral data cube corresponding to the region inside the rectangle. We use 188 spectral bands of this data set (low-SNR bands were dropped); thus, the used hyperspectral cube has the dimensions 350 × 350 × 188. In these experiments, we create our library $(\mathbf{D}^{188\times 325})$ from the USGS spectral library [50], which includes the spectra of the minerals present in the data cube.

The true abundance maps of the real-world data are not available; therefore, the previous works in sparse unmixing (e.g., [11] and [32]) use the real-world data for qualitative analysis. We follow the same approach and provide the results for visual comparison. We use the Tricorder classification map as a reference. Note that this reference map should not be considered as the ground truth for abundance maps. This is because a) the Tricorder map was computed in 1995, whereas the available data set was collected in 1997 by AVIRIS, and b) the Tricorder map considers each pixel to be pure and classifies it as a member of a class that is correlated to the representative mineral in the pixel. As unmixing performs subpixel classification, the computed results can be different from the classification results. Nevertheless, the Tricorder map serves as a good reference for the visual comparison of the sparse unmixing solutions.

Fig. 11 compares the proposed algorithms with the other popular unmixing algorithms for the abundance map estimation. The figure includes only those algorithms that showed good results in practical time. The parameter settings of all the algorithms are kept same as in Section VI-C3. From left to right, the first row of the figure shows the distribution maps of Tricorder for Alunite, Calcite, Dickite, and Chalcedony, respectively. The second and the third rows show the corresponding abundance maps computed by OMP-Star+ and OMP-Star, respectively. It is easy to see that these algorithms have generally computed high abundances at the pixels where the distribution maps show existence of the minerals. The other algorithms have also shown similar results; however, their estimated fractional abundances generally have lower values than those of the proposed algorithms. Results of the greedy algorithms in the figure were obtained in the same manner as those obtained in Tables I and II.

VIII. Complexity Analysis

If m is the number of spectral bands, k is the number of spectra in $\mathbf{D}$, and p is the cardinality of the mixed pixel, then the per-pixel computational complexity for the proposed algorithms is lower bounded by $\mathcal{O}(mkp)$ and upper bounded by $\mathcal{O}(Emkp)$. Here, E is the product of the parameter f and the cardinality of the set $\mathcal{T}$, averaged over the algorithm iterations. The lower bound is achieved when $|\mathcal{T}| = 1$ in all the iterations of the algorithm. In that case, the algorithm does not use the $ForwardSelection$ method at all. Therefore, its computational complexity becomes the same as OMP, which is $\mathcal{O}(mkp)$ [42]. The upper bound is achieved when $|\mathcal{T}| > 1$ in all the iterations. In general, we give the computational complexity of the proposed algorithms as $\mathcal{O}(emkp)$, where $1 \leq e \leq E$. In the experiments discussed in the earlier sections, e was on the order of 4 for the proposed algorithms. We illustrate this calculation with the help of Fig. 12. The figure shows the value of $|\mathcal{T}|$ for five iterations of OMP-Star, per pixel, for a data cube of 30 mixed pixels. We chose $t = 0.92$, and the cardinality of the mixed pixels is set to 5. The figure shows that $|\mathcal{T}| = 1$ for 63 iterations out of the total 150 iterations performed by the algorithm. Each of these 63 iterations has the same computational complexity as that of an OMP iteration. For the remaining 87 pixels, the average value of $|\mathcal{T}|$ is 3.42. With $f = 2$, the algorithm performs

[6] Available online: http://aviris.jpl.nasa.gov/html/aviris.freedata.html

[7] Available online: http://speclab.cr.usgs.gov/PAPERS/tetracorder/

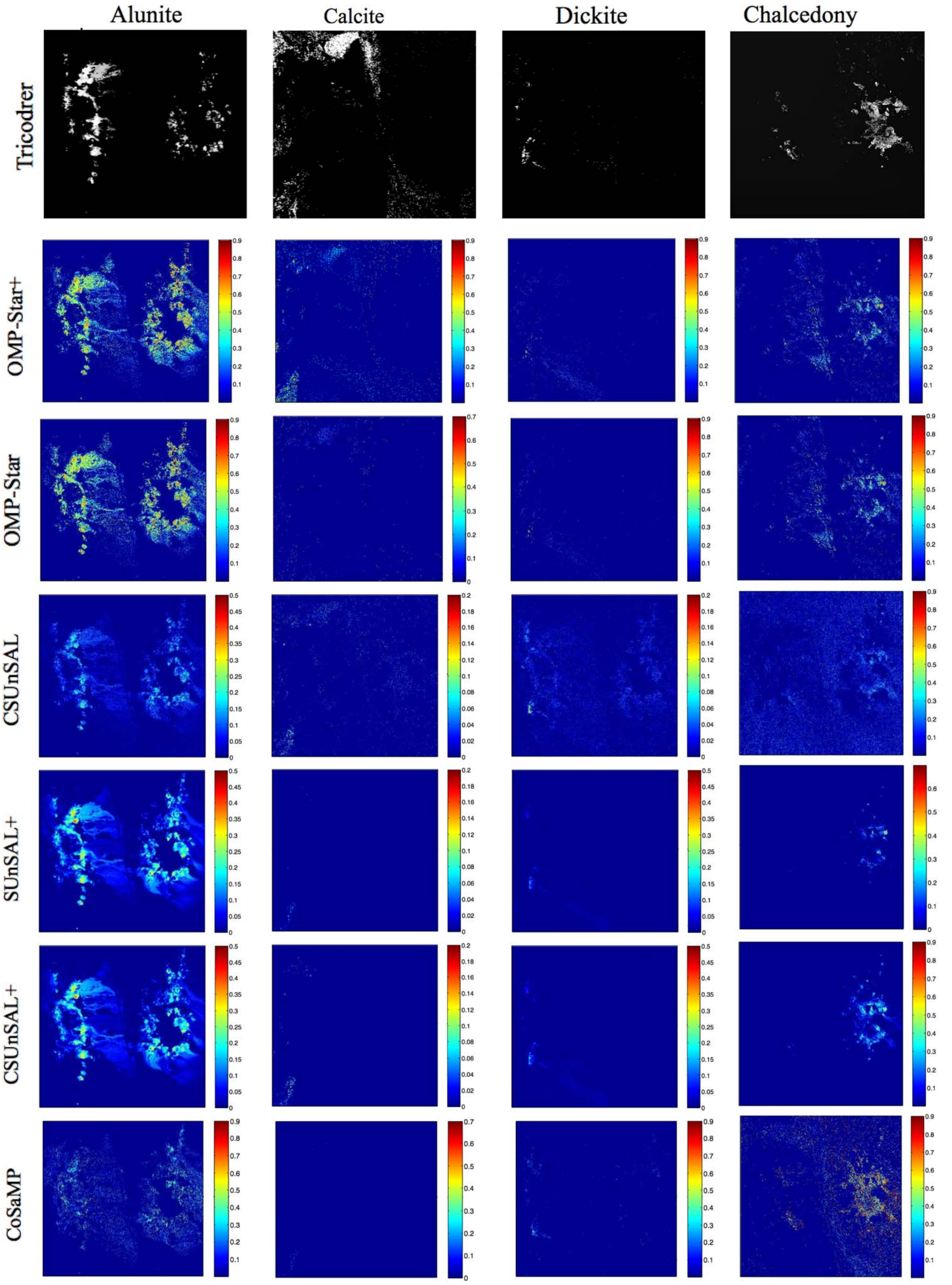


Fig. 11. Qualitative comparison of fractional abundance maps: The top row shows the distribution maps of Tricorder software for the 350 × 350 pixel AVIRIS Cuprite scene shown in the red boundary in Fig. 10. From left to right, the maps correspond to Alunite, Calcite, Dickite, and Chalcedony, respectively. Each row shows the fractional abundance maps calculated by a sparse unmixing algorithm. From top to bottom, the rows correspond to OMP-Star+, OMP-Star, CSUnSAL, SUnSAL+, CSUnSAL+, and CoSaMP, respectively.

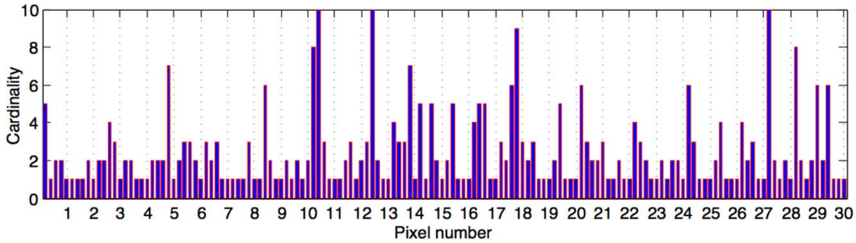


Fig. 12. Example of typical iterations of OMP-Star: The algorithm operates on a data cube of 30 mixed pixels with known $p = 5$. For each pixel, the algorithm performs five iterations. The five bars for each pixel show the cardinality of $\mathcal{T}$ for each iteration. For $t = 0.92$, $|\mathcal{T}| = 1$ for 63 out of the total 150 iterations.

TABLE III
PROCESSING TIME (IN SECONDS) FOR UNMIXING 500-PIXEL DATA CUBE WITH 35-dB SNR USING WHITE NOISE. EACH PIXEL IS A MIXTURE OF FIVE RANDOMLY SELECTED MATERIALS. TIME IS COMPUTED ON A DESKTOP PC EQUIPPED WITH AN INTEL CORE i7-2600 CPU (AT 3.4 GHz) AND 8-GB RAM

Algorithms	A*OMP	CSUnSAL+	SUnSAL+	CSUnSAL	LAOMP	SUnSAL	OMP-Star+
Time (s)	3812.8	49.9	48.8	33.5	32.5	31.4	19.4
Algorithms	OMP-Star	CoSaMP	OMP+	OMP	ROMP	gOMP	SP
Time (s)	14.4	3.9	3.1	2.8	1.9	1.7	1.6

a total of $87 \times 3.42 \times 2 \approx 595$ OMP-like iterations for these pixels. Thus, OMP-Star has the computational complexity of $63mkp + 595mkp = 658mkp$ for the whole data. Which gives $e = 658/150 \approx 4.4$ for the complete data cube.

The rest of the greedy algorithms used in this paper show the worst case computational complexities of the same order as OMP [38], [42], except LAOMP and A$*$OMP. LAOMP has the complexity $\mathcal{O}(mkp^2L)$, where L is the number of columns used in the *identification* step of the algorithm (see Section III). The computational complexity of A$*$OMP depends upon the chosen cost function, path pruning strategy, and related parameter settings. See [51] for details. The computational complexity of the convex relaxation-based algorithms used in this paper has been reported $\mathcal{O}(k^2)$ per iteration [35]. Table III compares the processing times of MATLAB implementations of all the algorithms used in this paper. The timings were computed for a hyperspectral data cube of 500 pixels with SNR $=35$ dB using white noise and $p = 5$. Except for the proposed algorithms, we assumed prior knowledge of the value of p for all the greedy algorithms. The proposed algorithms use the stopping rule (c) in line “17” of Algorithm 2, with $\beta = 0.9$. $t = 0.92$, and $f = 2$. Furthermore, the proposed algorithms use the four-step strategy, as mentioned in Section VI-D, to estimate the fractional abundances. Where required, we use the `lsqnonneg` procedure of MATLAB to implement the nonnegative least squares method. All the algorithms utilize the same parameter settings, as discussed in the previous sections.

IX. CONCLUSION

This paper proposes a greedy pursuit algorithm, called OMP-Star, for sparse unmixing of hyperspectral data. OMP-Star is a pixel-based algorithm that uses a futuristic greedy approach. This approach is inspired by a popular search algorithm, called A-Star. OMP-Star shows robustness against the high coherence of the data. This work also enhances OMP-Star to its nonnegative variant, called OMP-Star+. This constrained version of the algorithm exploits the fact that the fractional abundances of endmembers are nonnegative quantities. We propose to preprocess the hyperspectral data for greedy algorithms by taking its derivative. Generally, the derivative operation reduces the correlation among the spectral signatures, thereby improving the accuracy of SA results. However, this operation is sensitive to noise. Therefore, we explicitly evaluate derivatives for sparse unmixing and devise a strategy to use them with greedy algorithms. We test the proposed approach thoroughly on simulated and real-world hyperspectral data. The results demonstrate high effectiveness of the proposed approach.

APPENDIX A

Here, we show that the results from Theorem 1 in Section II-C are in line with Elad’s results in [31]. We follow the line of reasoning in [31] and arrive at the condition of equivalency of these results.

Proof: Consider the model

$$\mathbf{D}\boldsymbol{\alpha} = \mathbf{y}, \quad \text{s.t. } \boldsymbol{\alpha} > \mathbf{0}. \tag{23}$$

Since all the entries in $\mathbf{D}$ (i.e., reflectances) are nonnegative, the row span of $\mathbf{D}$ intersects the positive orthant. Mathematically,

$$\exists\, \mathbf{h} \text{ s.t. } \mathbf{h}^T\mathbf{D} = \mathbf{w}^T > \mathbf{0}. \tag{24}$$

Let $\mathbf{W} = \mathrm{diag}(\mathbf{w})$, which is a strictly positive definite matrix, then (23) can be rewritten as

$$\mathbf{D}\mathbf{W}^{-1}\mathbf{W}\boldsymbol{\alpha} = \mathbf{y}, \quad \text{s.t. } \boldsymbol{\alpha} > \mathbf{0} \tag{25}$$

or

$$\hat{\mathbf{D}}\mathbf{z} = \mathbf{y}, \qquad \mathbf{z} > \mathbf{0} \tag{26}$$

where $\hat{\mathbf{D}} = \mathbf{D}\mathbf{W}^{-1}$ and $\mathbf{z} = \mathbf{W}\boldsymbol{\alpha}$. Left multiplying (23) by $\mathbf{h}^T$ on both sides gives $c = \mathbf{h}^T\mathbf{y}$ on the right-hand side. Since $\mathbf{h}^T\mathbf{D} = \mathbf{w}^T$, we have $\mathbf{w}^T\boldsymbol{\alpha} = c$. That means

$$\mathbf{w}^T\boldsymbol{\alpha} = \mathbf{1}^T\mathbf{W}\boldsymbol{\alpha} = \mathbf{1}^T\mathbf{z} = c \tag{27}$$

where $\mathbf{1} \in \mathbb{R}^m$ is a column vector of 1s. Equation (27) suggests that the l_1-norm of $\mathbf{z} \in \{\hat{\mathbf{D}}\mathbf{z} = \mathbf{y}, \mathbf{z} > \mathbf{0}\}$ is the constant c.

In the aforementioned equations, we can always choose $\mathbf{h} = \mathbf{1}$, which makes $\hat{\mathbf{D}}$ nothing but $\mathbf{D}$ with its columns normalized in l_1-norm and makes $c = \|\mathbf{y}\|_1$. Thus, if the columns of $\mathbf{D}$ already have unit l_1-norm, then $\mathbf{W}$ is an identity matrix and $\mathbf{z} = \boldsymbol{\alpha}$. Therefore, $\|\mathbf{z}\|_1 = \|\boldsymbol{\alpha}\|_1 = \|\mathbf{y}\|_1$. This, in turn, implies automatic imposition of the generalized ASC, where $\|\mathbf{y}\|_1$ becomes the pixel-dependent scale factor. Note that this result coincides with Corollary 2a in Section II-C, and we arrive here after assuming the normalized version of $\mathbf{D}$.

It is easy to see that there can be many potential $\mathbf{h}$ vectors that satisfy the condition in (24). Which implies the existence of many potential matrices for $\mathbf{W}$, and many potential values of c. Therefore, c becomes a constant only when a particular $\mathbf{h}$ is operated on $\mathbf{D}$ to get $\mathbf{w}$. In [31], Elad used $\mathbf{h} = \mathbf{1}$ to convert the systems of equations in (23) to (26) and claimed $\|\mathbf{z}\|_1$ to be constant. No claim for $\|\boldsymbol{\alpha}\|_1$ to be constant was made in [31].

REFERENCES

[1] J. Bioucas-Dias *et al.*, "Hyperspectral remote sensing data analysis and future challenges," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 2, pp. 6–36, Jun. 2013.

[2] G. Camps-Valls, D. Tuia, L. Gomez-Chova, S. Jimenez, and J. Malo, *Remote Sensing Image Processing*. San Rafael, CA, USA: Morgan & Claypool, 2011.

[3] K. Staenz, A. Mueller, A. Held, and U. Heiden, "Technical committees corner: International spaceborne imaging spectroscopy (ISIS) technical committee," *IEEE Geosci. Remote Sens. Lett.*, vol. 165, pp. 38–42, 2012.

[4] J. M. Bioucas-Dias *et al.*, "Hyperspectral unmixing overview: Geometrical, statistical, sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observ.*, vol. 5, no. 2, pp. 354–379, 2012.

[5] J. Boardman, "Automating spectral unmixing of AVIRIS data using convex geometry concepts," in *Proc. Summaries 4th Annu. JPL Airborne Geosci. Workshop*, 1993, vol. 1, pp. 11–14.

[6] T. Chan, C. Chi, Y. Huang, and W. Ma, "Convex analysis based minimum volume enclosing simplex algorithm for hyperspectral unmixing," *IEEE Trans. Signal Process.*, vol. 57, no. 11, pp. 4418–4432, Nov. 2009.

[7] C. I. Chang, C. C. Wu, W. Liu, and Y. C. Ouyang, "A new growing method for simplex based endmember extraction algorithm," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 10, pp. 2804–2819, Oct. 2006.

[8] M. D. Craig, "Minimum-volume transforms for remotely sensed data," *IEEE Trans. Geosci. Remote Sens.*, vol. 32, no. 3, pp. 542–552, May 1994.

[9] A. Plaza and C. I. Chang, "Impact of initialization on design of endmember extraction algorithms," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 11, pp. 3397–3407, Nov. 2006.

[10] A. Plaza, P. Martinez, R. Perez, and J. Plaza, "A quantitative and comparative analysis of endmember extraction algorithms from hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 3, pp. 650–663, Mar. 2004.

[11] M. D. Iordache, J. Bioucas-Dias, and A. Plaza, "Sparse unmixing of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 6, pp. 2014–2039, Jun. 2011.

[12] J. M. P. Nascimento and J. Bioucas-Dias, "Vertex component analysis: A fast algorithm to unmix hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 4, pp. 898–910, Apr. 2005.

[13] T. H. Chan, W. K. Ambikapathi, and C. Y. Chi, "A simplex volume maximization framework for hyperspectral endmember extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4177–4193, Nov. 2011.

[14] M. E. Winter, "N-FINDR: An algorithm for fast autonomous spectral end-member determination in hyperspectral data," in *Proc. SPIE Imag. Spectrometry V*, 2003, vol. 3753, pp. 266–275.

[15] R. A. Neville, K. Staenz, T. Szeredi, J. Lefebvre, and P. Hauff, "Automatic endmember extraction from hyperspectral data for mineral exploration," in *Proc. 21st Can. Symp. Remote Sens.*, 1999, pp. 21–24.

[16] N. Gillis and S. Vavasis, "Fast and robust recursive algorithms for separable nonnegative matrix factorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 4, pp. 698–714, Apr. 2012.

[17] J. W. Boardman, F. A. Kruse, and R. O. Green, "Mapping target signatures via partial unmixing of AVIRIS data," in *Proc. JPL Airborne Earth Sci. Workshop*, 1995, pp. 23–26.

[18] J. Li and J. M. Bioucas-Dias, "Minimum volume simplex analysis: A fast algorithm to unmix hyperspectral data," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2008, pp. III-250–III-253.

[19] L. Miao and H. Qi, "Endmember extraction from highly mixed data using minimum volume constrained nonnegative matrix factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 3, pp. 765–777, Mar. 2007.

[20] M. Berman *et al.*, "ICE: A statistical approach to identify endmembers in hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 10, pp. 2085–2095, Oct. 2004.

[21] A. Zare and P. Gader, "Sparsity promoting iterated constrained endmember detection for hyperspectral imagery," *IEEE Geosci. Remote Sensing Lett.*, vol. 4, no. 3, pp. 446–450, Jul. 2007.

[22] M. Iordache, "A sparse regression approach to hyperspectral unmixing," Ph.D. thesis, Instituto Superior Tecnico, Universidade Tecnica de Lisboa, Lisboa, Portugal, Nov. 2011.

[23] T. T. Cai and L. Wang, "Orthogonal matching pursuit for sparse signal recovery with noise," *IEEE Trans. Inf. Theory*, vol. 57, no. 7, pp. 4680–4688, Jul. 2011.

[24] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, 1998.

[25] D. M. Rogge, B. Rivard, J. Zhang, and J. Feng, "Iterative spectral unmixing for optimizing per-pixel endmember sets," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 12, pp. 3725–3736, Dec. 2006.

[26] M. D. Iordache, J. 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.

[27] 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.

[28] X. Zhao, F. Wang, T. Huang, M. Ng, and R. Plemmons, "Deblurring and sparse unmixing for hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 7, pp. 4045–4058, Jul. 2013.

[29] R. Green *et al.*, "Imaging spectroscopy and the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS)," *Remote Sens. Environ.*, vol. 65, no. 3, pp. 227–248, Sep. 1998.

[30] B. Natarajan, "Sparse approximation solution to linear systems," *SIAM J. Comput.*, vol. 24, no. 2, pp. 227–234, Apr. 1995.

[31] A. Bruckstein, M. Elad, and M. Zibulevsky, "On the uniqueness of nonnegative sparse solutions to underdetermined systems of equation," *IEEE Trans. Inf. Theory*, vol. 54, no. 11, pp. 4813–4820, Nov. 2008.

[32] Z. Shi, W. Tang, Z. Duren, and Z. Jiang, "Subspace matching pursuit for sparse unmixing of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 6, pp. 3256–3274, Jun. 2014.

[33] M. Elad, *Sparse and Redundant Representation: From Theory to Application in Signal and Image Processing*. New York, NY, USA: Springer-Verlag, 2010.

[34] J. M. Bioucas-Dias and M. A. T. Figueiredo, "Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing," presented at the 2nd IEEE Workshop Hyperspectral Image Signal Processing, Evolution Remote Sensing, Jun. 2010.

[35] J. Eckstein and D. Bertsekas, "On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Math. Programm.*, vol. 55, no. 3, pp. 293–318, Jun. 1992.

[36] R. Tibshirani, "Regression shrinkage and selection via LASSO," *J. R. Stat. Soc.*, vol. 58, no. 1, pp. 267–288, 1996.

[37] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, pp. 3397–3415, Dec. 1993.

[38] J. Wang, S. Kwon, and B. Shim, "Generalized orthogonal matching pursuit," *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6202–6216, Dec. 2012.

[39] S. Chatterjee, D. Sundman, and M. Skoglund, "Look ahead orthogonal matching pursuit," in *Proc. IEEE ICASSP*, May 2011, pp. 4024–4027.

[40] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Appl. Comput. Harmonic Anal.*, vol. 26, no. 3, pp. 301–321, May 2009.

[41] D. Needell and R. Vershynin, "Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 310–316, Apr. 2010.

[42] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Trans. Inf. Theory*, vol. 55, no. 5, pp. 2230–2249, May 2009.

[43] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Trans. Syst. Sci. Cybern.*, vol. 4, no. 2, pp. 100–107, Jul. 1968.

[44] S. J. Russell, P. Norvig, J. F. Candy, J. M. Malik, and D. D. Edward, *Artificial Intelligence: A Modern Approach*. Upper Saddle River, NJ, USA: Prentice-Hall, 1996.

[45] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2231–2242, Oct. 2004.

[46] F. Tsai and W. Philpot, "Derivative analysis of hyperspectral data," *Remote Sens. Environ.*, vol. 66, no. 1, pp. 41–51, Oct. 1998.

[47] J. A. Tropp, A. C. Gilbert, and M. J. Strauss, "Algorithms for simultaneous sparse approximation: Part I: Greedy Pursuit," *Signal Process.*, vol. 86, no. 3, pp. 572–588, Mar. 2006.

[48] N. Akhtar, F. Shafait, and A. Mian, "Repeated constraint sparse coding with partial dictionaries for hyperspectral unmixing," in *Proc. IEEE WACV*, 2014, pp. 953–960.

[49] J. Bieniarz, R. Meuller, X. Zhu, and P. Reinartz, "On the use of overcomplete dictionaries for spectral unmixing," presented at the 4th Workshop Hyperspectral Image Signal Processing, Evolution Remote Sensing, 2012.

[50] R. N. Clark *et al.*, *USGS digital spectral library splib06a*. Denver, CO, USA: US Geological Survey, 2007.

[51] N. B. Karahanoglu and H. Erdogan, "Compressed sensing signal recovery via forward-backward pursuit," *Digit. Signal Process.*, vol. 23, no. 5, pp. 1539–1548, Sep. 2013.

[52] J. M. Bioucas-Dias and J. M. P. Nascimento, "Hyperspectral subspace identification," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 8, pp. 2435–2445, Aug. 2008.

Naveed Akhtar received the B.E. degree with distinction in aviation electronics from the National University of Sciences and Technology (NUST), Islamabad, Pakistan, in 2007 and the M.Sc. degree with distinction in autonomous systems from the Hochschule Bonn-Rhein-Sieg (HBRS), Sankt Augustin, Germany, in 2012. He is currently working toward the Ph.D. degree with The University of Western Australia (UWA), Crawley, W.A., Australia, under the supervision of Dr. Shafait and Prof. Mian.

From 2007 to 2009, he was a Research Assistant with the Research Institute for Microwaves and Millimeter-Wave Studies, NUST. In 2012, he was a Research Associate with the Department of Computer Science, HBRS. His current research is focused on hyperspectral image analysis, with emphasis on sparse representation-based techniques.

Mr. Akhtar was a recipient of the competitive scholarship for higher studies by Higher Education Commission, Pakistan, in 2009, and currently, he is a recipient of the Scholarship for International Research Fees scholarship at UWA.

Faisal Shafait received the Ph.D. degree in computer engineering with the highest distinction from the Kaiserslautern University of Technology, Kaiserslautern, Germany, in 2008.

He is currently an Assistant Professor with the School of Computer Science and Software Engineering, The University of Western Australia, Crawley, W.A., Australia. Formerly, he was a Senior Researcher at the German Research Center for Artificial Intelligence (DFKI), Germany, and a Visiting Researcher at Google, California, USA. His research interests include machine learning and pattern recognition, with a special emphasis on applications in document image analysis. He has coauthored over 80 publications in international peer-reviewed conferences and journals in this area.

Dr. Shafait is an Editorial Board Member of the *International Journal on Document Analysis and Recognition* (IJDAR) and a Program Committee Member of leading document analysis conferences, including International Conference on Document Analysis and Recognition, Document Analysis Systems, and International Conference on Frontiers in Handwriting Recognition. He is also serving on the Leadership Board of IAPR's Technical Committee on Computational Forensics (TC-6).

Ajmal Mian received the B.E. degree in avionics from the Nadirshaw Edulji Dinshaw (NED) University, Karachi, Pakistan, in 1993; the M.S. degree in information security from the National University of Sciences and Technology, Islamabad, Pakistan, in 2003; and the Ph.D. degree in computer science with distinction from The University of Western Australia, Crawley, W.A., Australia, in 2006.

He is currently a Research Professor with the School of Computer Science and Software Engineering, The University of Western Australia, where he has secured four national competitive research grants. His research interests include computer vision, pattern recognition, machine learning, multimodal biometrics, and hyperspectral image analysis.

Prof. Mian was a recipient of the Australasian Distinguished Doctoral Dissertation Award from the Computing Research and Education Association of Australia (CORE) in 2007. He was a recipient of two prestigious fellowships, including the Australian Postdoctoral Fellowship in 2008 and the Australian Research Fellowship in 2011. He was named the West Australian Early Career Scientist of the Year 2012.