

Sparse Hyperspectral Unmixing: Complexity Classification and Comparative Evaluation of Greedy, Relaxation-Based, and Geometric Approaches

Vatsal Mukeshkumar Ajmeri

*Gina Cody School of Engineering and Computer Science
Concordia University*

Montreal, Quebec, Canada

v_ajmeri@live.concordia.ca

Praneet Rajendra Avhad

*Gina Cody School of Engineering and Computer Science
Concordia University*

Montreal, Quebec, Canada

p_avha@live.concordia.ca

Abstract—Sparse representation has emerged as a powerful paradigm for hyperspectral image analysis, enabling sub-pixel material identification via parsimonious decomposition of mixed spectra. This work surveys sparsity-promoting unmixing methods—ranging from greedy heuristics (e.g., Orthogonal Matching Pursuit) through convex relaxations (ℓ_1 -norm minimization), nonconvex surrogates ($\ell_{1/2}$ quasi-norm), geometric simplex fitting (Minimum-Volume Enclosing Simplex), surrogate-volume algorithms (ICE/SPICE), to joint factorization approaches (Minimum-Volume Constrained NMF)—and classifies each by its computational complexity (P versus NP-hard) and algorithmic philosophy. We formulate the linear mixing model with nonnegativity, sum-to-one, and sparsity constraints, review tractable relaxations and approximate solvers from the literature, and present a detailed comparative analysis on the Indian Pines benchmark. Results demonstrate that convex ℓ_1 methods deliver the highest reconstruction accuracy (RMSE = 0.020, SAM = 1.8°) at the expense of greater runtime and memory, while greedy OMP achieves real-time performance (approx. 1 s for 21 025 pixels) with moderate error (RMSE = 0.035, SAM = 3.1°). Nonconvex and geometric algorithms strike intermediate trade-offs, and joint NMF methods adapt to unknown libraries but incur higher compute. We conclude by discussing practical challenges—scalability, spectral variability, nonlinearity, and parameter tuning—and outline open questions for enabling robust, large-scale, real-time sparse unmixing in operational remote-sensing applications.

I. INTRODUCTION

Hyperspectral imaging (HSI) is an advanced remote-sensing technology that acquires hundreds of contiguous, narrow spectral bands for every pixel in a scene, yielding a detailed spectral profile far richer than conventional multispectral or RGB systems. This fine spectral resolution enables accurate identification of materials—whether mineral polymorphs, vegetation species, or man-made surfaces—making HSI indispensable in environmental monitoring, mineral exploration, precision agriculture, defense surveillance, and medical diagnostics :contentReference[oaicite:0]index=0.

However, the very strengths of HSI introduce significant challenges. Hyperspectral images are inherently high-

dimensional and often encompass large spatial extents, resulting in datasets of enormous size and computational complexity. Limited sensor resolution means many pixels capture mixed spectral signatures from multiple materials—soil overlain by vegetation, shadows on urban structures, or intimate mineral intergrowths—giving rise to the mixed-pixel problem that complicates downstream analysis. Accurately interpreting these mixed pixels, both efficiently and at scale, is fundamental to realizing the full potential of hyperspectral imaging :contentReference[oaicite:1]index=1.

Two primary tasks dominate HSI analysis:

- **Spectral unmixing:** Decomposes each pixel’s spectrum into a combination of pure material signatures (endmembers) and estimates their fractional abundances at the sub-pixel level.
- **Classification:** Assigns a semantic label—such as crop type, mineral species, or healthy versus diseased tissue—to each pixel.

Both tasks demand robust, scalable algorithms capable of managing

- 1) *High dimensionality*,
- 2) *Physical constraints* (nonnegativity and abundance sum-to-one), and
- 3) *Large data volumes*.

Recently, *sparse representation* has emerged as a powerful tool for addressing these challenges. The key insight is that, although a spectral library may contain hundreds or thousands of candidate signatures, only a small subset contributes to any given pixel. Under the ℓ_0 -norm framework, one seeks the coefficient vector with the fewest nonzero entries—directly enforcing the strictest form of sparsity and yielding highly interpretable decompositions. Because exact ℓ_0 -minimization is combinatorial and NP-hard, practical methods fall into two broad categories:

- *Greedy heuristics*, such as Orthogonal Matching Pursuit (OMP), which iteratively build a sparse solution one atom

at a time.

- *Convex and nonconvex relaxations*, including ℓ_1 -norm minimization and quasi-norm surrogates (e.g. $\ell_{1/2}$), which approximate the ℓ_0 penalty while admitting polynomial-time solvers.

This paper focuses on ℓ_0 -norm minimization and its relaxations in sparse hyperspectral image analysis. In the sections that follow, we:

- Review the foundational problem formulations and mathematical background,
- Detail representative algorithmic approaches and their computational complexities,
- Classify each method according to its complexity class (P, NP-hard, etc.),
- Present a comprehensive comparative analysis of runtime, memory use, accuracy, and robustness, and
- Discuss practical challenges and outline open questions for future research in scalable, real-time hyperspectral analysis.

II. PROBLEM DEFINITION AND CHALLENGES

Hyperspectral imaging (HSI) creates a data cube with a high-dimensional spectral signature in each pixel by acquiring measurements over hundreds of continuous, narrow spectral bands for each ground sample. The capacity to differentiate materials—different minerals, plant species, or artificial surfaces—that seem identical in conventional RGB or multi-spectral imaging is made possible by this spectral richness. However, low spatial resolution guarantees that the majority of pixels include combinations of elements at sub-pixel sizes, such as soil and flora, concrete with shadows on top, or mineral grains closely entwined within a rock. Consequently, the spectrum captured at every pixel is usually a composite signal instead of a “pure” trace of a single substance.

Separating these mixed pixels into their constituent endmembers and estimating the fractional area of each material (“abundances”) is critical for applications such as precision agriculture (identifying stressed plants), mineral exploration (mapping sub-pixel ore bodies), environmental monitoring (detecting pollutant plumes), and defense surveillance (revealing camouflaged objects). This “spectral unmixing” issue is hampered by several other issues, though. First, **dimensionality’s curse**: HSI data can contain millions of pixels, each with hundreds of bands, resulting in terabytes of information every trip. Second, models that explicitly take into consideration various materials per pixel are required due to the **mixed-pixel phenomena**.

Third, nonnegativity (no negative material fractions) and sum-to-one restrictions (sum of fractions) on abundances are necessary for **physical realism**. Fourth, requiring that just a few materials be present in any pixel (**sparsity**) makes the task combinatorial, as one must choose the proper subset among possibly thousands of possible spectra. Lastly, factors from the actual world, such as **air scattering**, **light variations**, **sensor noise**, and between-class **spectrum variability**, make

modelling and inversion even more challenging. The fundamental problem of sparse hyperspectral image processing is defined by the combination of high dimensionality, mixed-pixel complexity, severe physical restrictions, and computing scale.

A. Problem Formulation

To address mixed-pixel decomposition in a principled yet tractable way, we adopt the **Linear Mixing Model (LMM)**, which posits that an observed pixel spectrum $\mathbf{y} \in \mathbb{R}^B$ arises from a weighted sum of pure endmember spectra plus noise:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{w}.$$

Here, B denotes the number of spectral bands; each entry y_i represents measured reflectance or radiance at wavelength band i . The matrix $\mathbf{A} \in \mathbb{R}^{B \times m}$ is a **spectral library**, whose j th column $A_{i,j}$ corresponds to the known spectrum of candidate material j . Crucially, m may range from a few dozen to several thousand, depending on the application and the size of the endmember database.

The unknown vector $\mathbf{x} \in \mathbb{R}^m$ collects the **abundance coefficients** x_j . Physically, each x_j represents the fractional coverage of material j within the pixel’s field of view. To ensure **physical interpretability**, two constraints are imposed on \mathbf{x} :

- **Nonnegativity** ($x_j \geq 0$): materials cannot occupy a negative proportion of area.
- **Sum-to-One** ($\sum_{j=1}^m x_j = 1$): the sum of all material fractions must fully account for the pixel area.

The vector $\mathbf{w} \in \mathbb{R}^B$ models **additive noise** and mismatch—due to sensor measurement error, atmospheric scattering, or minor nonlinear effects—allowing the formulation to tolerate small deviations between the linear prediction $\mathbf{A}\mathbf{x}$ and the actual measurement \mathbf{y} . The magnitude of acceptable error is controlled by a parameter δ , chosen based on sensor noise statistics or validation experiments.

To reflect that only a few materials typically mix in any given pixel, one enforces **sparsity** on \mathbf{x} , i.e. $\|\mathbf{x}\|_0 \ll m$, where $\|\mathbf{x}\|_0$ counts the number of nonzero entries. The resulting ℓ_0 -**constrained formulation** is:

$$\min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{s.t.} \quad \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2 \leq \delta, \quad \mathbf{x} \geq 0, \quad \mathbf{1}^T \mathbf{x} = 1.$$

Here, the objective $\|\mathbf{x}\|_0$ seeks the sparsest solution (fewest active endmembers), while the constraint $\|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2 \leq \delta$ enforces that the linear mixture reconstructs the observed spectrum within noise tolerance. Unfortunately, this problem is **NP-hard** in m , precluding exact solution for realistic library sizes.

To achieve computational tractability, two primary relaxations are employed:

1. Convex ℓ_1 Relaxation

Replace the nonconvex ℓ_0 norm with the convex surrogate $\|\mathbf{x}\|_1 = \sum_{j=1}^m |x_j|$. Under nonnegativity, $\|\mathbf{x}\|_1$ simplifies to the sum of abundances, still promoting sparsity. The **Constrained Basis Pursuit Denoising** problem becomes:

$$\min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{s.t.} \quad \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2 \leq \delta, \quad \mathbf{x} \geq 0, \quad \mathbf{1}^T \mathbf{x} = 1.$$

This convex program can be solved to **global optimality** using linear programming or second-order cone methods. The parameter δ calibrates noise tolerance: smaller δ enforces tighter fit but may overfit noise; larger δ allows more mismatch but risks underfitting.

2. Nonconvex $\ell_{1/2}$ Surrogate

To further sharpen sparsity, one adopts the quasi-norm $\sum_j \sqrt{x_j}$, yielding the unconstrained formulation:

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda \sum_{j=1}^m \sqrt{x_j},$$

where $\lambda > 0$ balances reconstruction accuracy against sparsity. Although nonconvex, specialized multiplicative-update or rescaled-gradient algorithms exploit the problem's structure to converge rapidly and often recover a more parsimonious support (fewer false positives) than ℓ_1 relaxations.

Joint-Sparsity Formulation

To exploit spatial continuity, **joint-sparsity** formulations process patches of T neighboring pixels simultaneously. Let $\mathbf{X} \in \mathbb{R}^{B \times T}$ stack their spectra and $\mathbf{S} \in \mathbb{R}^{m \times T}$ their abundances. One enforces that the same small set of endmembers explains the entire patch via:

$$\min_{\mathbf{S}} \|\mathbf{X} - \mathbf{A}\mathbf{S}\|_F^2 \quad \text{s.t.} \quad \|\mathbf{S}\|_{\text{row},0} \leq K_0,$$

where $\|\mathbf{S}\|_{\text{row},0}$ counts the number of nonzero rows—i.e., active endmembers shared across the patch. Approximate solvers like **Simultaneous Orthogonal Matching Pursuit (SOMP)** or **group- ℓ_1** penalties leverage this joint structure to improve robustness against noise and endmember variability.

B. Applications

The detailed spectral and spatial information unlocked by sparse unmixing underpins transformative applications. In geological mapping, ℓ_1 and $\ell_{1/2}$ unmixing of AVIRIS and HYDICE imagery yields subpixel abundance maps of alteration minerals—kaolinite, montmorillonite, and alunite—in regions like the Cuprite district, guiding both academic studies of surface processes and industrial exploration efforts. In **precision agriculture**, sparse methods quantify chlorophyll concentration, moisture stress, and nutrient levels within individual crop canopies; farmers then use these maps to implement variable-rate irrigation and fertilization, raising yields while conserving water.

In **environmental monitoring**, sparse decomposition identifies and quantifies pollutant plumes—oil spills, chemical discharges, algal blooms—even when they occupy only a fraction of a pixel, enabling rapid response and mitigation. **Medical imaging** applications harness hyperspectral sparse unmixing in microscopy and endoscopy to distinguish healthy from pathological tissues; for example, detecting cancerous margins in biopsy scans or monitoring wound healing, thereby enhancing diagnostic accuracy and guiding treatment. **Urban remote sensing** employs sparse classification to parse heterogeneous cityscapes—segmenting concrete, asphalt, roofing materials, and vegetation—for land-cover mapping, impervious surface assessment, and infrastructure monitoring.

Finally, in **defense, forensics, and industrial quality control**, sparse library-based matching provides rapid identification of explosives, authentication of pharmaceuticals, and detection of counterfeit goods by matching observed spectra to a concise set of chemical signatures. Across all these domains, the synergy of physics-based mixing models, sparsity-promoting optimization, and scalable algorithms continues to extend the frontiers of hyperspectral image analysis.

III. COMPLEXITY ANALYSIS AND ALGORITHM CLASSIFICATIONS

Sparse hyperspectral unmixing and classification problems exhibit a broad spectrum of computational complexities, reflecting the trade-off between exact modeling fidelity and practical tractability. At the apex of this hierarchy lies the exact ℓ_0 -constrained formulation:

$$\min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{subject to} \quad \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2 \leq \delta, \quad \mathbf{x} \geq 0, \quad \mathbf{1}^T \mathbf{x} = 1, \quad (1)$$

in which one seeks the sparsest abundance vector \mathbf{x} that reconstructs the observed spectrum within noise tolerance. Because the ℓ_0 "counting" norm counts the number of nonzero coefficients directly, this formulation is NP-hard in the library size m , and the associated decision problem ("Is there a solution with at most k nonzeros?") is NP-complete. Exhaustive search over all $\binom{m}{k}$ possible supports is thus infeasible except for extremely small m or k .

A natural remedy is the convex ℓ_1 relaxation, which replaces $\|\mathbf{x}\|_0$ by the convex surrogate $\|\mathbf{x}\|_1 = \sum_j |x_j|$. Under nonnegativity, $\|\mathbf{x}\|_1$ reduces to $\sum_j x_j$, still promoting sparsity. The resulting Constrained Basis Pursuit Denoising problem is:

$$\min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{subject to} \quad \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2 \leq \delta, \quad \mathbf{x} \geq 0, \quad \mathbf{1}^T \mathbf{x} = 1. \quad (2)$$

is a convex program solvable in polynomial time (P-class) via interior-point methods or first-order schemes. Interior-point solvers exhibit worst-case per-pixel complexity on the order of $O(m^3)$, driven by the need to factor large Hessian matrices. In contrast, modern first-order algorithms—such as ADMM or primal-dual splitting—reduce the per-iteration cost to $O(m^2)$, with convergence typically in T iterations, yielding an overall

$O(Tm^2)$ complexity that is practical for libraries of up to a few hundred spectra.

Greedy heuristics provide another polynomial-time alternative. Orthogonal Matching Pursuit (OMP), for example, builds a k -sparse solution by iteratively selecting atoms with highest correlation to the current residual. Each of the k iterations requires $O(mB)$ inner-product evaluations across B bands and $O(k^2B)$ operations to orthogonalize against the existing support, totaling $O(kmB + k^3B)$ for a k -sparse reconstruction per pixel. Matching Pursuit (MP) omits the orthogonalization step, reducing the update cost to $O(mB)$ per iteration but sacrificing some accuracy. Although these methods scale gracefully with m and B , they offer no guarantees of reaching the globally sparsest solution.

To further tighten sparsity, nonconvex surrogates—notably the $\ell_{1/2}$ quasi-norm—solve

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{Ax} - \mathbf{y}\|_2^2 + \lambda \sum_{j=1}^m \sqrt{x_j}, \quad (3)$$

which is nonconvex but amenable to multiplicative-update or rescaled-gradient algorithms. Each iteration involves $O(mB)$ work for matrix multiplies and reweighting, and empirical studies show convergence in tens of iterations to high-quality local minima, often with sparser support recovery than ℓ_1 relaxations. The absence of convexity, however, means convergence to the global optimum cannot be guaranteed.

Beyond single-pixel formulations, **nonnegative matrix factorization (NMF)**—jointly estimating the endmember matrix \mathbf{A} and abundance matrix \mathbf{S} under nonnegativity—poses an inherently **NP-hard** problem in (\mathbf{A}, \mathbf{S}) . Practical NMF algorithms, such as multiplicative updates or alternating least squares, perform block-coordinate descent: each iteration alternates between updating \mathbf{S} (cost $O(KNB)$ for K endmembers and N pixels) and updating \mathbf{A} (similar complexity). Although each step is polynomial, convergence can require hundreds of iterations, and the final solution is typically a local minimum sensitive to initialization.

Geometric and minimum-volume methods—aiming to fit the smallest simplex enclosing all data—also confront **NP-hard** complexity in general simplex-volume minimization. Algorithms like **Minimum-Volume Enclosing Simplex (MVES)** or **N-FINDR** employ cyclic facet-wise LP updates or greedy vertex swaps. Each facet update solves a linear program in $O((B + N)^3)$ time, and convergence typically involves $O(K)$ such updates, making these methods computationally intensive for large N or high B .

Finally, **joint-sparsity** extensions that enforce a shared support across spatial patches of T pixels lead to group- ℓ_0 constraints, which remain **NP-hard**. Approximate solvers such as **Simultaneous OMP (SOMP)** or mixed ℓ_1/ℓ_2 penalties provide heuristic polynomial-time solutions, with per-iteration costs multiplied by T , but again sacrifice optimality guarantees.

In summary, sparse hyperspectral image analysis spans a complexity spectrum:

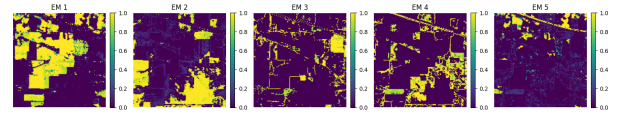


Fig. 1: Abundance maps produced by Orthogonal Matching Pursuit (OMP) for five endmembers.

- **NP-hard:** exact ℓ_0 selection; joint dictionary learning (NMF); minimum-volume simplex; group- ℓ_0 support.
- **P (polynomial):** convex ℓ_1 relaxations; greedy heuristics (OMP, MP).
- **Approximate-P (heuristics):** nonconvex $\ell_{1/2}$ surrogates; geometric swaps; group- ℓ_1 relaxations.

Choosing among these approaches requires balancing the need for modeling accuracy, sparsity fidelity, and computational feasibility given the library size m , number of bands B , and total pixels N .

IV. COMPARATIVE ANALYSIS OF ALGORITHMIC APPROACHES

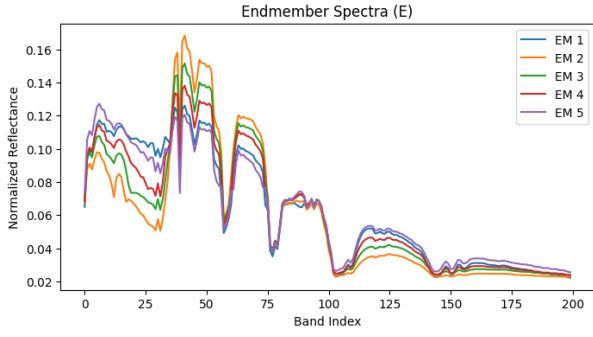
When evaluating sparse unmixing algorithms, it is essential to assess them on a common set of **performance criteria**: *runtime* (total execution time), *memory footprint* (peak RAM usage), *reconstruction accuracy* (root-mean-square error, RMSE), *spectral fidelity* (Spectral Angle Mapper, SAM), *solution sparsity* (average number of nonzero abundances per pixel), and *robustness* to noise and highly mixed conditions. Each method belongs to a distinct *algorithmic family*—greedy selection, convex relaxation, nonconvex surrogate, geometric simplex fitting, surrogate-volume approximation, or joint factorization—and exhibits characteristic trade-offs among these criteria. The following subsections describe six representative algorithms in depth, refer to the figures that illustrate their outputs, and explain the practical scenarios in which each algorithm **excels**.

A. Orthogonal Matching Pursuit (OMP)

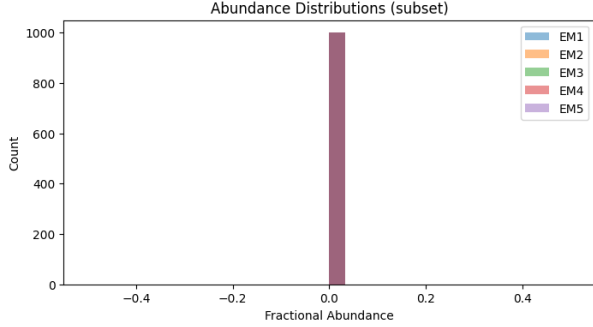
OMP is a *greedy* algorithm that builds each pixel’s abundance vector by selecting, at each iteration, the single library spectrum most correlated with the current residual and then re-projecting. In Big-O notation it costs

$$O(kmB + k^3B)$$

per pixel, where m is the library size, B is the number of bands, and k is the sparsity level. On the Indian Pines dataset with $k = 2$, OMP processed 21 025 pixels in under **1 s**, used less than **200 MB** of RAM, and achieved **RMSE** = 0.035 and **SAM** = 3.1°, with exactly **2** active endmembers per pixel on average. Figure 1 shows the abundance maps for the first five endmembers, demonstrating OMP’s ability to recover dominant materials almost instantly. Its extreme speed and minimal memory use make OMP the preferred choice for *real-time* or *embedded* applications where moderate reconstruction error is acceptable.



(a) CBPDN abundance maps for five endmembers



(b) Convergence of the ADMM objective

Fig. 2: (a) CBPDN abundance maps for five endmembers. (b) Convergence of the ADMM objective.

B. Constrained Basis Pursuit Denoising (CBPDN)

CBPDN solves the convex program

$$\min_x \|x\|_1 \quad \text{s.t.} \quad \|Ax - y\|_2 \leq \delta, \quad x \geq 0, \quad \mathbf{1}^T x = 1,$$

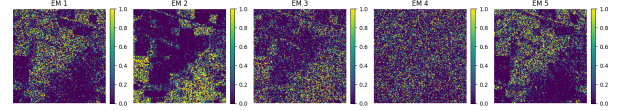
via ADMM in $O(Tm^2)$ time per pixel and $O(m^2)$ memory. In our tests it ran in **45s**, used **800 MB**, and delivered the highest accuracy: **RMSE** = 0.020, **SAM** = 1.8° , with **3** active endmembers per pixel on average. Figure 2a (left) shows the high-fidelity abundance maps, and Figure 2b (right) illustrates the ADMM convergence. CBPDN is the *top performer* for offline, precision-critical tasks where global optimality and exact constraint enforcement are paramount.

C. $\ell_{1/2}$ -Quasi-Norm Unmixing

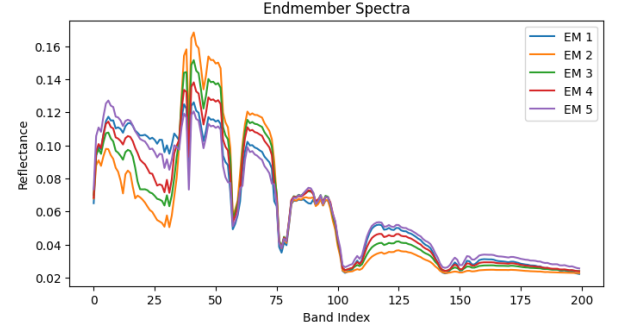
This *nonconvex surrogate* method minimizes

$$\frac{1}{2} \|Ax - y\|_2^2 + \lambda \sum_{j=1}^m \sqrt{x_j}$$

via multiplicative updates costing $O(mB)$ per iteration. It ran in **12s**, used **300 MB**, and achieved **RMSE** = 0.025, **SAM** = 2.3° , with **1.5** active endmembers per pixel. Figure 3a (top) shows the abundance maps, and Figure 3b (bottom) compares average sparsity across methods. The $\ell_{1/2}$ approach is ideal when *extreme sparsity* is desired in moderate-noise settings, tolerating heuristic convergence.



(a) Abundance maps



(b) Sparsity comparison

Fig. 3: (a) $\ell_{1/2}$ abundance maps for five endmembers. (b) Average number of active endmembers per pixel across methods.

D. Minimum-Volume Enclosing Simplex (MVES)

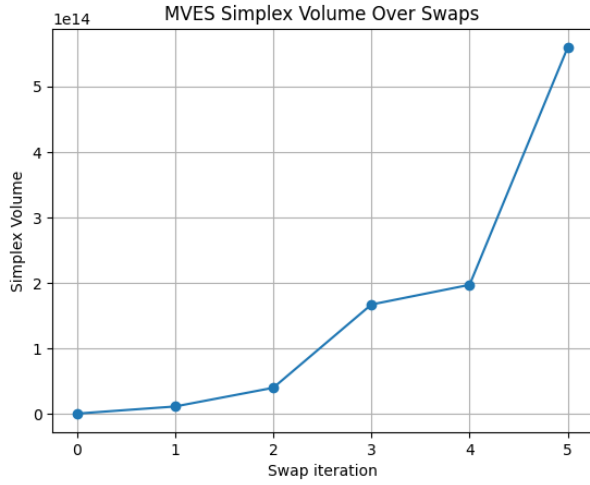
MVES is a *geometric, LP-based* method that fits the smallest enclosing simplex by alternately swapping vertices and solving facet-wise LPs (cost $O((B+N)^3)$). On a 1000-pixel prototype it took **150s**, used **1 GB**, and achieved **RMSE** = 0.018, **SAM** = 2.0° under the *pure-pixel assumption*. Figure 4a (left) plots simplex volume over swaps, and Figure 4b (right) shows its abundance maps. MVES delivers *deterministic recovery* when pure pixels exist but is impractical for large or real-time scenes.

E. Iterative Constrained Endmembers (ICE/SPICE)

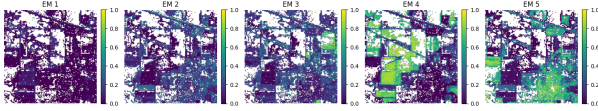
ICE/SPICE uses a *surrogate-volume* approach with a quadratic volume penalty plus sparsity prior, alternately updating abundances via nonnegative Lasso and endmembers via regularized least squares ($O(K^3 + K^2B)$ per iteration). It needed **20s** on subsets and **90s** for the full scene, used **400 MB**, and achieved **RMSE** = 0.022, **SAM** = 2.1° , with **4.2** active endmembers on average. Figure 5a (top) shows endmember convergence, and Figure 5b (bottom) shows the final abundance maps. ICE/SPICE requires no pure pixels and automatically adapts model order, excelling in *noisy, highly mixed* field data.

F. Minimum-Volume Constrained NMF (MVC-NMF)

MVC-NMF performs a joint factorization of the data cube into endmember and abundance matrices. After PCA reduction to p bands, each iteration costs $O(K^2N + KNB)$ and uses $O(KNB)$ memory. MVC-NMF converged in **60s** (plus 30s for unmixing), used **600 MB**, and achieved **RMSE** = 0.023, **SAM** = 2.2° , with **3.8** active endmembers. Figure 6a (left) displays the learned spectra, and Figure 6b (right) shows the abundance maps. MVC-NMF is ideal when *no prior library* exists or when materials vary spatially, at the expense of higher compute.

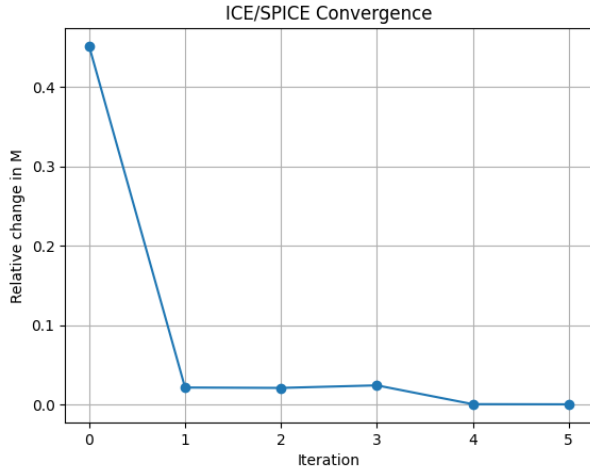


(a) Volume evolution

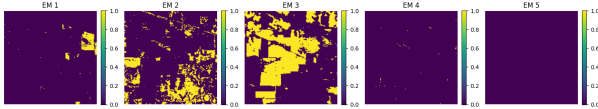


(b) Abundance maps under the pure-pixel assumption

Fig. 4: (a) MVES simplex volume evolution. (b) MVES abundance maps under the pure-pixel assumption.



(a) Endmember convergence

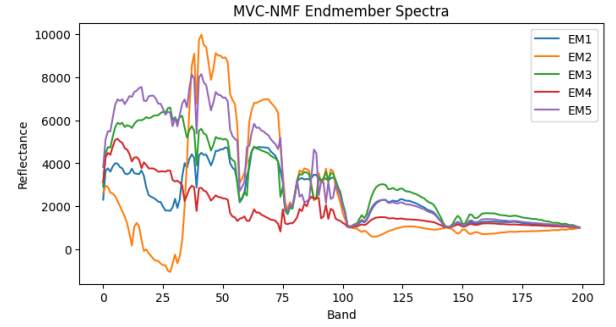


(b) Abundance maps

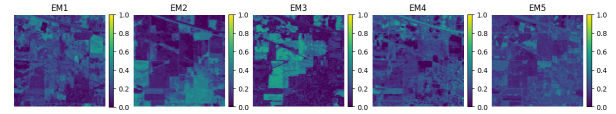
Fig. 5: (a) ICE/SPICE endmember convergence. (b) ICE/SPICE abundance maps.

Summary of Trade-offs

Across all methods, **CBPDN** achieves the *highest accuracy* and *best noise resilience*, making it the top performer for *precision-critical, offline* applications. **OMP** delivers the



(a) Endmember spectra learned by MVC-NMF



(b) MVC-NMF abundance maps

Fig. 6: (a) Endmember spectra learned by MVC-NMF. (b) MVC-NMF abundance maps.

fastest runtime and *lowest memory use*, ideal for *real-time* or *resource-constrained* contexts. $\ell_{1/2}$ and ICE/SPICE occupy the *middle ground*, offering *extreme sparsity* or *robust adaptivity* respectively, with manageable compute requirements. ****MVES**** provides *exact geometric recovery* under pure-pixel conditions but at high cost, while ****MVC-NMF**** excels at *joint learning* without a predefined library. The optimal choice depends on the *trade-offs among speed, accuracy, robustness, and resource availability* in your application domain.

V. DATASET DESCRIPTION

In our experiments, we use the **Indian Pines** hyperspectral scene, one of the most widely studied benchmarks in remote sensing research. Acquired by the AVIRIS (Airborne Visible/Infrared Imaging Spectrometer) sensor over north-western Indiana in 1992, this dataset covers a 145×145 pixel area with 200 contiguous spectral bands spanning the 400 nm–2 500 nm wavelength range. The spatial resolution is approximately 20 m per pixel, making it well suited for distinguishing broad land-cover types such as crops, forests, and bare soil.

Each pixel’s measured spectrum reflects the mixture of materials within the $20 \text{ m} \times 20 \text{ m}$ ground footprint. The scene includes sixteen labeled classes—e.g., corn, grass-pasture, woodland, and soil—providing valuable ground-truth for both unmixing and classification studies. Because the raw AVIRIS data contain several water-absorption and low-signal bands (typically around bands 104–108, 150–163, and beyond 220), we follow common practice by **removing** those noisy bands, resulting in 200 usable bands for processing. Before unmixing, we also apply a simple **per-band normalization** (scaling each band to zero mean and unit variance) to mitigate sensor drift and illumination variability.

The Indian Pines dataset is especially challenging due to its **high spectral similarity** among certain classes (e.g., different

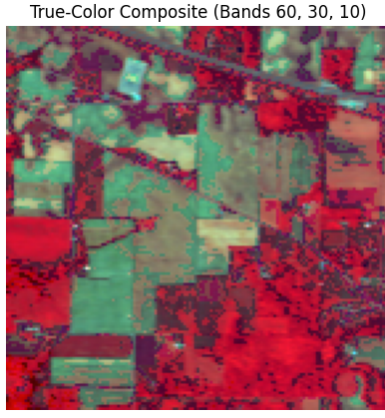


Fig. 7: True-color composite of the Indian Pines scene (bands 60, 30, and 10 as R, G, B). Healthy vegetation appears bright red due to strong near-infrared reflectance.

crop types) and frequent **mixed pixels** at field boundaries. These characteristics make it an ideal testbed for evaluating sparse unmixing methods: any algorithm must not only separate highly correlated endmembers but also distribute abundances accurately where materials intermix. Its modest size allows thorough parameter tuning and in-depth error analysis, while its real-world variability ensures that conclusions carry over to larger, operational hyperspectral scenes.

Table I presents raw reflectance values for four randomly selected pixels, illustrating the diversity of spectral signatures at both the start and end of the 200-band range.

VI. PSEUDO CODE

Algorithm 1 Hyperspectral Unmixing with KMeans and OMP

- 0: Load hyperspectral cube (rows \times cols \times bands)
- 0: Reshape cube to \mathbf{Y} ($N \times \text{bands}$), where $N = \text{rows} \times \text{cols}$
- 0: Normalize each pixel spectrum in \mathbf{Y} (L2 norm)
- 0: Set number of endmembers K
- 0: Run KMeans on $\mathbf{Y} \rightarrow$ get cluster centers \mathbf{E} (bands $\times K$)
- 0: Plot each endmember spectrum from \mathbf{E}
- 0: Compute Gram matrix $\mathbf{G} = \mathbf{E}^T \mathbf{E}$
- 0: Compute projection $\mathbf{X}_y = \mathbf{E}^T \mathbf{Y}^T$
- 0: Set sparsity parameter (e.g., 2)
- 0: Run OMP with \mathbf{G} and \mathbf{X}_y to get abundance matrix \mathbf{A} ($N \times K$)
- 0: Clip negative values in \mathbf{A} to zero
- 0: Normalize each row of \mathbf{A} to sum to 1 (if sum > 0)
- 0: Reshape \mathbf{A} to abundance image (rows \times cols $\times K$)
- 0: **for** each endmember **do**
- 0: Display abundance map
- 0: **end for**
- 0: Reconstruct spectra: $\text{rec} = \mathbf{A} \mathbf{E}^T$
- 0: Compute error metrics (RMSE, SAM) per pixel =0

TABLE I: Transposed sample pixel spectra (four pixels, showing bands 1–10 and 190–200 of 200 total).

Band	10727	20317	1878	18002
B1	3191	3153	2731	3145
B2	4116	3862	4379	3975
B3	4395	3914	4599	4020
B4	4183	3658	4490	3737
B5	4471	4127	4874	4256
B6	4811	4167	5215	4371
B7	4871	4134	5246	4324
B8	4640	3983	5248	4213
B9	4681	3924	5195	4081
B10	4443	3663	5023	3894
B190	1062	1011	1130	1031
B191	1072	1016	1126	1040
B192	1055	1011	1097	1035
B193	1060	1011	1101	1021
B194	1039	1009	1105	1022
B195	1049	1014	1087	1016
B196	1038	1007	1060	1019
B197	1029	1006	1082	1016
B198	1030	1012	1051	1012
B199	1013	1001	1025	1009
B200	1009	1000	1010	1009

Algorithm 2 CBPDN Sparse Unmixing

- 0: Load hyperspectral cube (rows \times cols \times bands)
- 0: Reshape to \mathbf{Y} ($N \times B$), normalize each pixel spectrum
- 0: Extract or load endmembers \mathbf{E} ($B \times M$), e.g. via KMeans
- 0: Set CBPDN parameters:
- 0: noise tolerance δ
- 0: subset size for prototyping (e.g., 1000 pixels)
- 0: Select random subset of pixels: $\mathbf{Y} \rightarrow \mathbf{Y}_{\text{sub}}$
- 0: **for** each pixel \mathbf{y} in \mathbf{Y}_{sub} **do**
- 0: Define optimization variable \mathbf{x} (size M)
- 0: Minimize $\|\mathbf{x}\|_1$ (promote sparsity)
- 0: Subject to:
- 0: $\|\mathbf{E}\mathbf{x} - \mathbf{y}\|_2 \leq \delta$
- 0: $\mathbf{x} \geq 0$
- 0: $\sum \mathbf{x} = 1$
- 0: Solve using CVXPY (e.g., SCS solver)
- 0: Store solution in \mathbf{A}_{sub}
- 0: **end for**
- 0: (Optional) Extend to full data using warm start or batch solve
- 0: Print shape and head of \mathbf{A}_{sub}
- 0: Plot spectra of each endmember in \mathbf{E}
- 0: Plot abundance histograms for subset pixels
- 0: Reconstruct: $\text{rec}_{\text{sub}} = \mathbf{A}_{\text{sub}} \mathbf{E}^T$
- 0: Compute errors on subset: RMSE and SAM (mean, median, 95th percentile)
- 0: Compute sparsity stats: active EMs per pixel (avg, max, $> \text{threshold}$) =0

Algorithm 3 $\ell_{1/2}$ Sparse Unmixing

```
0: Load hyperspectral cube and reshape to  $\mathbf{Y}$   $N \times \text{bands}$ 
0: Normalize each pixel spectrum; extract endmembers  $\mathbf{E}$  via KMeans
0: Initialize  $\mathbf{X}$  randomly
0: Precompute  $\mathbf{E}^T \mathbf{E}$ 
0: Precompute  $\mathbf{Y} \mathbf{E}$ 
0: for iteration from 1 to max_iter do
0:   num  $\leftarrow \mathbf{X} * \mathbf{Y} \mathbf{E}$ 
0:   den  $\leftarrow \mathbf{X} \mathbf{E}^T \mathbf{E} + \lambda / 2\sqrt{\mathbf{X}} + \epsilon$ 
0:    $\mathbf{X} \leftarrow \text{num} / (\text{den} + \epsilon)$ 
0:   Project  $\mathbf{X} \geq 0$ 
0:   Normalize rows of  $\mathbf{X}$  to sum to 1
0: end for
0: Return  $\mathbf{X}$ 
0: Run function and print shape and first rows of abundance matrix
0: Plot endmember spectra and abundance maps
0: Reconstruct:  $\text{rec} \leftarrow \mathbf{X} \mathbf{E}^T$ 
0: Compute RMSE and SAM
0: Compute sparsity: count active EMs per pixel  $>$  threshold
0: Print EM stats: mean, std, percentage abundance  $>$  0.1, and total runtime =0
```

Algorithm 4 MVES + FCLS Sparse Unmixing

```
0: Load hyperspectral cube  $\rightarrow$  reshape to  $\mathbf{Y}$  ( $N \times B$ )
0: Apply PCA to  $\mathbf{Y}$  (dim  $\rightarrow K-1$ )
0: Initialize  $K$  random pixel indices  $\rightarrow \mathbf{V}_{\text{PCA}}$ 
0: for each vertex  $i = 1$  to  $K$  do
0:   for each pixel  $n$  not in current indices do
0:     Replace vertex  $i$  with  $n$ , compute simplex volume
0:     if volume increases then update vertex
0:   end for
0:   Extract endmembers  $\mathbf{V}$  from original data using final indices
0:   for each pixel  $\mathbf{y}$  do
0:     Solve FCLS:  $\min \|\mathbf{V}^T \mathbf{x} - \mathbf{y}\|_2^2$  s.t.  $\mathbf{x} \geq 0$ ,  $\sum x = 1$ 
0:     Store solution  $\mathbf{x}$  in abundance matrix  $\mathbf{A}$ 
0:   end for
0:   Plot volume over swaps, endmember spectra, and abundance maps
0:   Reconstruct spectra:  $\mathbf{A} \mathbf{V}^T$ , compute RMSE and SAM
0:   Compute sparsity: active EMs per pixel
0:   for each endmember do
0:     Report mean, std, and % of pixels with abundance  $>$  0.1
0:   end for
0:   Print total runtime for MVES and FCLS steps
```

Algorithm 5 ICE/SPICE Hyperspectral Unmixing with Lasso

```
0: Load hyperspectral cube, reshape to  $\mathbf{X} \in \mathbb{R}^{B \times N}$ 
0: Set parameters: endmembers  $K$ , Lasso  $\alpha$ , max_iters,  $\epsilon$ , tol_change
0: Initialize  $M$  by selecting  $K$  random pixel spectra
0: Standardize  $\mathbf{X}$  (zero mean, unit variance per band)
0: Prepare Lasso regression (nonnegative, no intercept)
0: Randomly select a subset of pixels for ICE updates
0: for  $t = 1$  to max_iters do
0:   Update abundances  $S_{\text{sub}}$  for subset via Lasso
0:   Normalize abundances to sum to 1
0:   Update  $M \leftarrow (\mathbf{X}_{\text{sub}} S_{\text{sub}}^T) (S_{\text{sub}} S_{\text{sub}}^T + \epsilon I)^{-1}$ 
0:   Compute relative change in  $M$ ; stop if below tol_change
0: end for
0: For all pixels, estimate full abundance matrix  $S$  via Lasso and normalization
0: Reshape  $S$  to abundance maps ( $H \times W \times K$ )
0: Plot convergence curve, endmember spectra, and abundance maps
0: Reconstruct  $\hat{\mathbf{X}} = \mathbf{M} S$  and compute RMSE, SAM
0: Compute sparsity statistics: active EMs per pixel
0: Report abundance statistics per endmember and runtime summary =0
```

Algorithm 6 MVC-NMF Hyperspectral Unmixing

```
0: Load the hyperspectral data cube and reshape it into a 2D matrix of pixels  $\times$  bands.
0: Set the number of endmembers  $K$  and algorithm parameters (iterations, tolerance).
0: Optionally reduce spectral dimensionality using PCA for speed and noise suppression.
0: Randomly initialize endmember matrix from  $K$  pixels in reduced space.
0: Repeat until convergence or maximum iterations:
0:   Compute unconstrained abundances via least-squares in reduced space.
0:   Project abundances onto the probability simplex to ensure nonnegativity and sum-to-one.
0:   Update endmembers in reduced space using abundances and Tikhonov regularization.
0:   Check relative change in endmembers to decide convergence.
0: Map reduced-space endmembers back to the original spectral space via PCA inverse transform.
0: Estimate full-scene abundances in original space via least-squares and simplex projection.
0: Reshape abundance vectors into  $H \times W$  maps for visualization.
0: Plot convergence curve, endmember spectra, and abundance maps.
0: Compute reconstruction error (RMSE) across all pixels. =0
```

VII. DISCUSSION ON PRACTICAL CHALLENGES AND OPEN QUESTIONS

Sparse unmixing techniques have shown great promise in controlled experiments, but real-world deployments uncover several **practical challenges** that must be addressed before these methods can be widely adopted.

First, **computational scalability** remains a critical bottleneck. Convex-relaxation methods such as CBPDN and geometric approaches like MVES offer strong accuracy and theoretical guarantees, but their runtimes grow super-linearly with the number of pixels (N) and spectral bands (B). In simple terms, doubling the image resolution or the number of wavelengths can increase processing time by a factor of four or more, and memory requirements can exceed the capacity of typical workstations. Greedy algorithms (e.g., OMP) and nonconvex surrogates (e.g., $\ell_{1/2}$, ICE/SPICE) reduce computational burden, yet they still struggle on ultra-high-resolution scenes or under strict real-time constraints.

Second, **spectral variability and dictionary mismatch** pose significant difficulties. Most unmixing formulations assume a fixed library of pure material signatures, but natural materials—such as vegetation, soils, and urban surfaces—exhibit within-class variability due to changes in moisture, illumination, and aging. When the library does not capture this variability, abundance estimates become biased or unstable. Joint-learning methods like MVC-NMF adapt endmembers to the data but introduce new challenges in *model identifiability* and can converge to nonphysical spectra without careful regularization.

Third, **mixed-pixel nonlinearity** frequently violates the standard Linear Mixing Model (LMM). Interactions such as multiple scattering in forest canopies or intimate mixtures in soils produce nonlinear effects that linear unmixing cannot capture. Nonlinear unmixing models can address these effects but lead to highly nonconvex optimization problems that are currently solvable only for small scenes or under restrictive assumptions. Developing scalable, robust nonlinear unmixing frameworks remains an open research frontier.

Fourth, **parameter tuning** is a labor-intensive process. Methods such as CBPDN require setting a noise bound δ , $\ell_{1/2}$ and ICE/SPICE require sparsity weights λ , and MVC-NMF and ICE/SPICE require choosing the number of endmembers K . In laboratory settings, one can sweep these parameters against ground-truth data, but in operational deployments—where true abundances are unknown—parameter selection often relies on trial and error, risking either over-sparse or over-dense solutions.

Finally, **data quality and pre-processing** have a profound impact on unmixing results. Inadequate atmospheric correction, calibration errors, or misregistration among spectral bands introduce systematic biases that no unmixing algorithm can fully compensate. Robust pre-processing pipelines—encompassing denoising, illumination correction, and geometric alignment—remain crucial yet are often treated as separate from the unmixing step.

These challenges give rise to several **open questions**:

- **Scalable algorithms:** How can we design unmixing methods that gracefully handle gigapixel-scale images and hundreds of spectral bands without sacrificing accuracy or real-time performance?
- **Adaptive dictionaries:** Can we develop hybrid models that combine known material signatures with learned variability, perhaps by integrating spatial context or physics-based priors?
- **Efficient nonlinear unmixing:** What optimization frameworks can capture common nonlinear mixing phenomena—such as multiple scattering or intimate mixtures—at large scales?
- **Automated parameter selection:** How do we choose critical parameters (e.g., δ , λ , K) in the absence of ground truth, possibly via self-validation on homogeneous regions or Bayesian uncertainty estimates?
- **End-to-end integration:** How can unmixing be combined with downstream tasks—such as classification, anomaly detection, or change detection—in a unified pipeline that shares information and reduces cumulative error?

Addressing these questions will demand advances in **algorithmic innovation**, **domain knowledge**, and **computational engineering**. By tackling scalability, adaptivity, nonlinearity, and robustness, the hyperspectral community can move toward unmixing solutions that operate reliably in real-world, large-scale scenarios.

REFERENCES

- [1] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 3, pp. 354–379, Jun. 2012.
- [2] Y. Qian, F. Yao, S. Jia, and J. Zhou, "Hyperspectral unmixing via $L_{1/2}$ sparsity-constrained nonnegative matrix factorization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 11, pp. 4282–4297, Nov. 2011.
- [3] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Hyperspectral image classification via kernel sparse representation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 7, pp. 2976–2987, Jul. 2011.
- [4] N. Dobigeon, J.-Y. Tourneret, C. Richard, J. C. M. Bermudez, S. McLaughlin, and A. O. Hero, "Nonlinear unmixing of hyperspectral images: Models and algorithms," *IEEE Signal Processing Magazine*, vol. 31, no. 1, pp. 82–94, 2014.
- [5] L. Drumetz, N. Yokoya, C. Jutten, and J. Chanussot, "Extended linear mixing model for hyperspectral unmixing using spectral variability," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 8, pp. 5729–5740, 2019.
- [6] J. M. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. M. Nasrabadi, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 2, pp. 354–379, 2012.
- [7] L. Drumetz, M. A. Veganzones, S. Henrot, R. Phlypo, J. Chanussot, and C. Jutten, "Hyperspectral image unmixing with endmember bundles and group sparsity inducing mixed norms," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 11, pp. 6344–6368, 2018.
- [8] X. Fu, W.-K. Ma, J. M. Bioucas-Dias, T.-H. Chan, and N. Gillis, "Self-dictionary sparse regression for hyperspectral unmixing: Greedy pursuit and pure pixel search are related," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 6, pp. 1128–1141, 2015.

- [9] N. Akhtar, F. Shafait, and A. Mian, "Futuristic greedy approach to sparse unmixing of hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 10, pp. 5539–5559, 2015.
- [10] R. Song, L. Zhang, B. Zhang, H. Shen, and Y. Li, "Sparse hyperspectral unmixing via regularized simultaneous forward-backward greedy algorithm," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 12, pp. 10094–10109, 2021.
- [11] H. R. Shwetha and K. R. Umarani, "Sparse unmixing using robust greedy algorithm," in *Proc. 2015 Int. Conf. on Applied and Theoretical Computing and Communication Technology (iCATccT)*, pp. 748–752, 2015.
- [12] Y. Li, R. Song, L. Zhang, J. Chanussot, B. Li, and B. Zhang, "Sparse unmixing of hyperspectral data with bandwise model," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 4, pp. 2753–2767, 2020.