IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 14, 2021, accepted March 1, 2021, date of publication March 4, 2021, date of current version March 12, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3063798

Sparse Analysis Recovery via Iterative Cosupport Detection Estimation

HEPING SONG[1,2], XIAOLONG REN[1], YUPING LAI[3], AND HONGYING MENG[4], (Senior Member, IEEE)

[1]School of Computer Science and Communications Engineering, Jiangsu University, Zhenjiang 212013, China
[2]Jiangsu Key Laboratory of Security Technology for Industrial Cyberspace, Jiangsu University, Zhenjiang 212013, China
[3]School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing 100876, China
[4]Department of Electronic and Electrical Engineering, Brunel University London, Uxbridge UB8 3PH, U.K.

Corresponding authors: Heping Song (songhp@ujs.edu.cn) and Yuping Lai (laiyp2009@126.com)

This work was supported in part by the Beijing Natural Science Foundation (BNSF) under Grant No. 4194076, the Natural Science Foundation of Jiangsu Province under Grant No. BK20170558 and the China Scholarship Council (CSC, No. 202008320094).

ABSTRACT Cosparse analysis model (CAM) provides a new signal processing paradigm for recovering cosparse signals with respect to a given analysis operator from the undersampled linear measurements in the context of emerging theory of compressed sensing (CS). The sparse analysis recovery/cosparse recovery is a key one brought up by this new paradigm. In this paper, we propose a new family of analysis pursuit algorithms for the sparse analysis recovery problem when the signals obey the cosparse analysis model, termed as iterative cosupport detection estimation (ICDE). ICDE is an algorithmic framework, which alternates between detecting a cosupport set of the unknown true signal and estimating the underlying signal by solving a truncated analysis pursuit problem on the detected cosupport. Further, we propose effective implementations of ICDE equipped with an efficient thresholding strategy for cosupport detection. Empirical performance comparisons show that ICDE is favorable in comparison with the state-of-the-art sparse analysis recovery algorithms. Source code of ICDE has been made publicly available on Github: https://github.com/songhp/ICDE.

INDEX TERMS Sparse representation, compressed sensing, sparse signal processing, cosparse analysis model.

I. INTRODUCTION

Data models for sparsity-exploiting applications in image and signal processing have drawn much attention in last decade [1], [2]. In the context, the sparse synthesis model (SSM) [3] offers an elegant approach to lead the era of sparse representation [4]–[8]. In this model, the unknown signal $\mathrm{x} \in \mathbb{R}^d$ of interest can be represented as a linear combination of some atoms of fixed matrix D (column vectors). The mathematical model of SSM can be denoted as $\mathrm{x} = D\mathrm{z}$, where $D \in \mathbb{R}^{d\times n}$ is a overcomplete dictionary ($d \leq n$), and $\mathrm{z} \in \mathbb{R}^n$ is the sparse representation of signal x. The *sparsity* $k = \|\mathrm{z}\|_0$ ($\|\cdot\|_0$ is a ℓ_0 pseudo-norm counting the total number of non-zero entries in a vector) assumed to be much smaller than n. The *support* is the index set of non-zero coefficients of z, which synthesize the signal x from atoms of D. The underlying signal x can be reconstructed by exploring the following optimization problem in the classical compressed sensing (CS) [9], [10] setting:

$$\hat{\mathrm{z}} = \arg\min_{\mathrm{z}} \|\mathrm{z}\|_0 \quad s.t. \quad \mathrm{y} = M\mathrm{x} = MD\mathrm{z}, \tag{1}$$

where $M \in \mathbb{R}^{m\times d}$ represents a known measurement matrix/operator, y is the linear measurements/observations, and $m \leq d$. Hence, the signal estimation $\hat{\mathrm{x}}$ is computed as $\hat{\mathrm{x}} = D\hat{\mathrm{z}}$. This issue can be expressed as recovering an unknown high-dimensional signal x from a limited set of measurements y of a low dimension. Since the problem (1) is computationally intractable, the optimization problem is relaxed to basis pursuit (BP) [11]

$$\hat{\mathrm{z}} = \arg\min_{\mathrm{z}} \|\mathrm{z}\|_1 \quad s.t. \quad \mathrm{y} = M\mathrm{x} = MD\mathrm{z}, \tag{2}$$

or other ℓ_1 regularizations. Another alternative solution for (1) is based on the idea of iterative greedy pursuit, such as orthogonal matching pursuit (OMP) [12], [13], subspace pursuit (SP) [14], compressive sampling matching pursuit

The associate editor coordinating the review of this manuscript and approving it for publication was Prakasam Periasamy.

(CoSaMP) [15], hard thresholding pursuit (HTP) [16] and recent work [17].

A. COSPARSE ANALYSIS MODEL

Recently, the "twin" model of the SSM that takes an analysis point of view, cosparse analysis model (CAM), was proposed in [3]. We consider the analysis CS problem. For a given analysis operator/dictionary $\Omega \in \mathbb{R}^{p\times d}$, $\mathrm{z} = \Omega\mathrm{x}$ is supposed to be sparse (i.e., contain adequately few non-zero coefficients). Although the two models become interchangeable as Ω is invertible (i.e. $\Omega = D^{-1}$) [18], we will mainly focus on non-invertible Ω, e.g., the "overcomplete" case where $p > d$. The analysis model is motivated not by a sparse representation of x in an overcomplete dictionary D, but rather by the sparsity of $\Omega\mathrm{x}$ for a given analysis operator Ω. The signal of interest is said to be *cosparse* with respect to the analysis operator Ω if the analysis coefficient $\Omega\mathrm{x}$ contains many zero entries. Further, the number of zeros $l = p - \|\Omega\mathrm{x}\|_0$ (i.e, the number of rows in Ω that are orthogonal to x) is called the **cosparsity** of x with respect to Ω, the **cosupport** Λ of x is defined by the index set of the zero entries of the analyzed vector $\Omega\mathrm{x}$, i.e., $\Lambda := \{i : (\Omega\mathrm{x})_i = 0\}$. Unlike SSM, CAM recovers the original signal x directly by performing optimization

$$\hat{\mathrm{x}} = \arg\min_{\mathrm{x}} \|\Omega\mathrm{x}\|_0 \quad s.t. \quad \mathrm{y} = M\mathrm{x}. \tag{3}$$

With relaxation by replacing ℓ_0 with ℓ_1 norm, resulting with the analysis basis pursuit (ABP) [19] or ℓ_1 analysis minimization [20]

$$\hat{\mathrm{x}} = \arg\min_{\mathrm{x}} \|\Omega\mathrm{x}\|_1 \quad s.t. \quad \mathrm{y} = M\mathrm{x}, \tag{4}$$

or generalized LASSO [21] in the Lagrangian form

$$\hat{\mathrm{x}} = \arg\min_{\mathrm{x}} \frac{1}{2}\|\mathrm{y} - M\mathrm{x}\|_2^2 + \lambda\|\Omega\mathrm{x}\|_1. \tag{5}$$

B. SPARSE ANALYSIS RECOVERY

Just as in the synthesis case, the important alternative option for recovering cosparse signals is used to a family of greedy-like algorithms. Nam *et al.* [3], [22] proposed greedy analysis pursuit (GAP) as an analysis counterpart corresponding to OMP. A modified GAP algorithm was developed to solve the weighted regularized L2-minimization problem, as proposed in Reference [23]. Mohagheghian *et al.* [24] proposed an enhanced weighted GAP (ewGAP), an extension to the GAP algorithm based on synthesis counterpart Lorentzian norm [25]. Cosparsity-based stagewise matching pursuit (CSMP) [26] employed more sophisticated backtracking methods as an analysis counterpart corresponding to stagewise orthogonal matching pursuit (StOMP) [27]. Giryes and Elad and Giryes *et al.* [28], [29] proposed a family of analysis pursuit algorithms which can be interpreted as a generalization of greedy methods to analysis regularizers of the form (3). Analysis SP (ASP), analysis CoSaMP (ACoSaMP) and analysis HTP (AHTP) are applied for the synthesis equivalent methods SP, CoSaMP and HTP, respectively.

As discussed above, one important direction to develop numerical methods for sparse analysis recovery is to employ synthesis counterpart algorithms. Borgerding *et al.* [30] developed a Bayesian approach to cosparse analysis CS based on the generalized approximate message passing (GAMP) algorithm. Xie *et al.* [31] proposed a new method based on accelerated alternating minimization, which can be interpreted as a generalization of synthesis counterpart L1 method. Co-IRW-L1 [32] can be interpreted as a generalization of iteratively reweighted L1 (IRW-L1) method to analysis regularizers of the form (5). Gong *et al.* [33] developed matching pursuit generalized LASSO (MPGL) method, which generalizes alternating direction method of multipliers (ADMM) [34] using matching pursuit [12].

C. CONTRIBUTION

Motivated by the aforementioned works, we develop a general class of analysis pursuit algorithms, termed iterative cosupport detection-estimation (ICDE), attempting to provide analysis versions of the synthesis counterpart algorithms. ICDE alternatively executes its two steps: cosupport detection and signal estimation. For an incorrect recovery, cosupport detection refines the index set Λ by pruning some elements of $\Omega\mathrm{x}$, and signal estimation solves the truncated ABP optimization

$$\hat{\mathrm{x}} = \arg\min_{\mathrm{x}} \|\Omega_\Lambda\mathrm{x}\|_1 \quad s.t. \quad \mathrm{y} = M\mathrm{x}. \tag{6}$$

From the new solution, cosupport detection can prune more elements in Λ and thus estimate a better $\hat{\mathrm{x}}$. In this wise, the two steps of ICDE iterate alternatively to gradually refine Λ and improve $\hat{\mathrm{x}}$. In summary, the major contributions of our work are as follows:

- A new algorithmic framework ICDE is provided for reconstructing the cosparse signals obey the sparse analysis model. Moreover, ICDE generalizes L1 norm (6) to a general cosparsity inducing regularizer.
- Effective implementations of ICDE equipped with an efficient thresholding strategy are proposed.
- The proposed algorithms have been evaluated and demonstrated that it can match or even outperform the state-of-the-art techniques that use a much larger number of measurements.

D. ORGANIZATION

The rest of this paper is organized as follows. In section II, after the introduction of synthesis counterpart, the proposed algorithmic framework of ICDE was presented along with two simple demos. The experimental evaluation of the proposed approaches was demonstrated in Section III. In Section IV, we conclude this article with discussions on the future of the work.

E. NOTATIONS AND PRELIMINARIES

The following notations are used throughout this work:

- $\hat{x}^{(t)}$: the algorithms presented in this work are iterative and the recovered signal $\hat{x}$ in current iteration t is denoted as $\hat{x}^{(t)}$. The same notation is used for other vectors and matrices.
- $|x|$, $\|x\|_{\ell_p}$, x^T: the absolute value, ℓ_p norm and transpose of a vector x, respectively. $\|x\|_1$ is the ℓ_1 norm that sums the absolute vector values and $\|x\|_2$ is the euclidian norm. $\|x\|_0$, though not rally a norm, is the ℓ_0 norm that counts the total number of non-zero entries of a vector.
- T, A_T: index set T, the matrix A_T refers to restricting rows from A indexed by T, whereas it has taken the columns in the synthesis case. The same convention is used for vectors.
- $A^\dagger$: the Moore-Penrose pseudoinverse of matrix $A \in \mathbb{R}^{m\times n}$. $A^\dagger = A^T(AA^T)^{-1}$ for $m \leq n$; $A^\dagger = (A^TA)^{-1}A^T$ for $m \geq n$.
- $[1, p]$ denotes the set of integers $\{1, 2, \cdots, p\}$.
- T^C, $[1, p] \setminus T$: the complement of set T in set $[1, p]$.
- supp(x): the support set of a vector x, i.e. the index set corresponding to the nonzeros of x, $\text{supp}(x) = \{i : x_i \neq 0\}$.
- cosupp(x). the cosupport set of a vector x with respect to analysis operator Ω, i.e. the index set corresponding to the zeros of Ωx, $\text{cosupp}(x) = \{i : (\Omega x)_i = 0\}$.
- $H(x, k)$: the hard thresholding operator that sets all but the k largest entries of a vector x to 0 for the synthesis cases.
- $H(\Omega x, l)$: the hard thresholding operator that sets the smallest l (in magnitude) entries of a vector Ωx to 0 for the analysis cases.

II. PROPOSED ANALYSIS PURSUIT ALGORITHMS

A. ITERATIVE SUPPORT DETECTION

Before introducing the analysis counterpart methods, we recall their synthesis versions. Since the failed signal recovery of BP, Wang and Yin [35] present an algorithmic framework to improve the BP constructions, called iterative support detection (ISD). Earlier work [36] developed a similar approach termed Iterative Detection-Estimation (IDE). ISD iterates between two steps: support detection and signal estimation. Starting from the detected support $I = \emptyset$ and the iteration number $t = 0$, ISD iterates between two main steps:

1) **Signal reconstruction:**
   solve the truncated BP problem with $T = I^C$:
   $\hat{x}^{(t)} = \arg\min_{x} \|x_T\|_1 \; s.t. \; y = Ax$;
2) **Support detection:**
   detect the support set I using $\hat{x}^{(t)}$ as the reference.

The reliability of ISD depends on the support detection, Wang and Yin proposed several detection strategies for different kinds of sparse signals. One of the general support detection strategies is based on thresholding

$$I^{(t)} = \{i : |\hat{x}_i^{(t)}| > \beta^t \; \max |\hat{x}^{(t)}|\}, \beta \in (0, 1). \tag{7}$$

Algorithm 1 ICDE-L1 Algorithm

Input: Measurement matrix M, analysis operator Ω, measurements y, thresholding parameter β.
Output: The reconstructed signal $\hat{x}$.

Initialization:
  $t = 1$ //iteration number
  $\Lambda^{(0)} = [1, p]$ //initial cosupport set
  $\hat{x}^{(0)} = \arg\min_x \|\Omega x\|_1 \;\; s.t. \;\; y = Mx$ //initial signal
while *halting criterion false* **do**
  $\Lambda^{(t)} = [1, p] \setminus \{i : z_i \geq \beta \max z_j, z = |\Omega\hat{x}^{(t)}|, i \in [1, p], j \in \Lambda^{(t-1)}, \beta \in (0, 1]\}$
  $\hat{x}^{(t)} = \arg\min_x \|\Omega_{\Lambda^{(t)}} x\|_1 \;\; s.t. \;\; y = Mx$
  $t = t + 1$
end while
return $\hat{x}^{(t)}$

B. ITERATIVE COSUPPORT DETECTION-ESTIMATION

Armed with the synthesis counterparts, we can "translate" each synthesis step into an analysis one and devise the analysis versions of the ISD. The proposed algorithmic framework describes as follows:
Input: measurement matrix M, analysis operator Ω, measurements y.
1. Initialize $\hat{x} = \mathbf{0}$ or other estimation and set the iteration number $t = 1$;
2. While the stopping criterion is not met, do
i) $\Lambda^{(t)} \leftarrow$ cosupport detection using Ωx as the reference;
ii) $x^{(t)} \leftarrow$ signal estimation by solving the truncated ABP (6) for $\Lambda = \Lambda^{(t)}$;
iii) $t \leftarrow t + 1$.

Like ISD, ICDE is an algorithmic framework. The implementation of ICDE depends on cosupport detection which requires an effective reference and an efficient detection strategy. We employ the gradient step of iterative thresholding algorithms as the reference. Iterative thresholding algorithms are simple and versatile sparse reconstruction methods, such as iterative soft thresholding (IST) [37], iterative hard thresholding (IHT) [38], and a large class of iterative shrinkage thresholding algorithms (ISTA) [39], [40]. As to signal estimation, the truncated least-square solution

$$\hat{x} = \arg\min_{x} \|\Omega_\Lambda x\|_2^2 \;\; s.t. \;\; y = Mx, \tag{8}$$

is much easier to solve than (6). The answer to (8) is obvious since $\hat{x}$ satisfies the linear equation [3]

$$\begin{bmatrix} y \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} M \\ \Omega_\Lambda \end{bmatrix} x. \tag{9}$$

The approximate analytical solution is

$$\hat{x} = \begin{bmatrix} M \\ \sqrt{\lambda}\Omega_\Lambda \end{bmatrix}^\dagger \begin{bmatrix} y \\ \mathbf{0} \end{bmatrix} = \left(M^TM + \lambda\Omega_\Lambda^T\Omega_\Lambda\right)^{-1} M^T y, \tag{10}$$

where λ is Lagrange multiplier. For high-dimensional signals it is efficient to solve by some numerical algorithms (e.g., the conjugate gradient method [29]). Therefore, we present two signal estimation methods, and term the resulting algorithms ICDE-L1 and ICDE-L2 respectively. We introduce the implementation of ICDE-L1, ICDE-L2 equipped with an efficient thresholding policy. The complete description of the proposed algorithms is presented here:

Step 1: Initialization:
Initialize the recovered signal $\hat{x}^{(0)}$,
initialize cosupport set $\Lambda^{(0)} = [1, p]$,
and set the iterative counter $t = 1$.

Step 2: Cosupport detection:
Update signal approximation:
$\hat{x}^{(t)} = \hat{x}^{(t-1)} + M^T(y - M\hat{x}^{(t-1)})$.
Detect the cosupport set $\Lambda^{(t)}$:
$\Lambda^{(t)} = [1, p] \setminus \{i : z_i \geq \beta \max z_j, z = |\Omega\hat{x}^{(t)}|, i \in [1, p], j \in \Lambda^{(t-1)}, \beta \in (0, 1]\}$.

Step 3: Signal estimation:
Estimate the signal $\hat{x}^{(t)}$ by solving (6) or (8) on the detected cosupport $\Lambda^{(t)}$.

Step 4: Halting:
Check whether the halting condition is False. If so, update $t = t + 1$ and back to Step 2. The detailed steps in implementation of ICDE-L1, ICDE-L2 are presented in Algorithm 1 and Algorithm 2 respectively. For ICDE-L1, we need ABP (4) solution as the initialization of $\hat{x}$ and the gradient step is redundant. Like ISD, ICDE is an algorithmic framework, which generalizes support detection in synthesis cases to cosupport detection for the analysis counterpart. Furthermore, ICDE adopts the two-stage thresholding [37], [41], [42] to solve the analysis pursuit problem. The first stage thresholding is the cosupport detection to make sure that the analyzed signal Ωx is sparse. the second stage thresholding employs the truncated L1/L2 optimization (6)/(8) to refine signal reconstruction. Hence, greedy-like analysis pursuit algorithms such as GAP, ASP and AHTP can be treated as special cases of ICDE. Comparing ASP/AHTP and ICDE/GAP, the former employ the hard thresholding to find the cosupport set Λ and demand the prior knowledge about the true cosparsity level l. It is not acceptable for the problems that the true cosparsity level l is not available. ASP and AHTP fix the cardinality of cosupport set Λ and remove previous false detections. GAP iteratively maintains a cosupport of gradual decreased indices by thresholding

$$\Lambda^{(t)} = \Lambda^{(t-1)} \setminus \{i : z_i \geq \beta \max z_i, z = |\Omega x^{(t-1)}|, \quad i \in \Lambda^{(t-1)}, \beta \in (0, 1]\}. \tag{11}$$

However, ICDE refines the cosupport set Λ that is not necessarily decreasing or nested over the iterations. This is useful since it is very hard thing to completely avoid false detections. ICDE and GAP directly detect the cosupport of the true cosparse signal by referencing the analyzed vectors Ωx while ASP and ACoSaMP augment the cosupport by handpicking the least values of the "correlation" coefficients $\Omega M^T(y - Mx)$.

We presented an ICDE-L1 demo that reconstructs a cosparse vector with $p = 220, d = 200, l = 190, m = 60, \beta = 0.5$ (detailed experimental setting in section III). In this setting, it is generally considered hard to reconstruct a cosparse signal with 30 non-zeros in Ωx from only 60 measurements. As depicted in the first subplot of Figure 1, ABP fails to reconstruct the cosparse signal due to insufficient measurements. However, ICDE-L1 exhibits an exact reconstruction after merely four iterations. Given an original cosparse signal x, if $\Lambda = \text{cosupp}(x)$, then the solution of (6) is, obviously, equal to x. Furthermore, exact recovery can happen even if Λ includes few spurious indices, as illustrated in the 3rd, 4th subplots of Figure 1, where the true analysis vectors Ωx are marked by black point and the recovered $\Omega\hat{x}$ are marked separately by red circle and blue star, denoting true and false non-zeros, respectively. Green lines represent the thresholds. As shown in the title of each subplot, we define the quadruplet "(total, good, bad, miss)" and "Err" as follows:

- total: the number of total nonzeros of the recovered $\Omega\hat{x}$, total = good + bad.
- good: the number of true nonzeros of the recovered $\Omega\hat{x}$.
- bad: the number of false nonzeros of the recovered $\Omega\hat{x}$.
- miss: the number of undetected nonzeros in the true Ωx, $p-l$ = good + miss.
- Err: the relative error $\|\Omega\hat{x} - \Omega x\|_2 / \|\Omega x\|_2$.

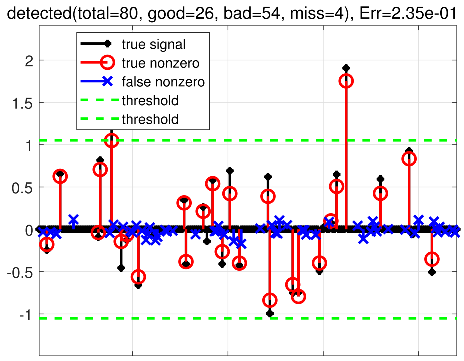
We now turn from ICDE-L1 demo to ICDE-L2 demo as shown in Figure 2. From the upper left subplot, different from ICDE-L1, ICDE-L2 detects a large number of "bad" coefficients. However, most of the "good" coefficients were relatively large in magnitude. The detection yielded new cosupport Λ which was sufficient to let (8) return a lot better solution $\hat{x}$ presented in the 2nd subplot. This solution further detected a tighter threshold to yield smaller "bad" coefficients. Particularly, most of the "good" coefficients with big magnitude had been exactly detected. The succeeding iterations[1] had exhibited better solutions that the "good" coefficients well matched the true signal and the "bad" coefficients were converging to zero. The final solu-

Algorithm 2 ICDE-L2 Algorithm

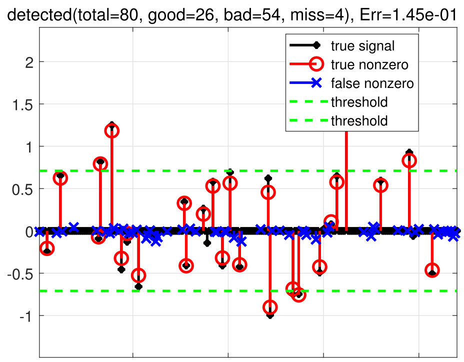
Input: Measurement matrix M, analysis operator Ω, measurements y, thresholding parameter β.
Output: The reconstructed signal $\hat{x}$.

Initialization:
$t = 1$ //iteration number
$\Lambda^{(0)} = [1, p]$ //initial cosupport set
$\hat{x}^{(0)} = \mathbf{0}$ //initial signal
while *halting criterion false* **do**
  $\hat{x}^{(t)} = \hat{x}^{(t-1)} + M^T(y - M\hat{x}^{(t-1)})$
  $\Lambda^{(t)} = [1, p] \setminus \{i : z_i \geq \beta \max z_j, z = |\Omega\hat{x}^{(t)}|, i \in [1, p], j \in \Lambda^{(t-1)}, \beta \in (0, 1]\}$
  $\hat{x}^{(t)} = \arg\min_x \|\Omega_{\Lambda^{(t)}} x\|_2^2 \quad s.t. \quad y = Mx$
  $t = t + 1$
end while
return $\hat{x}^{(t)}$

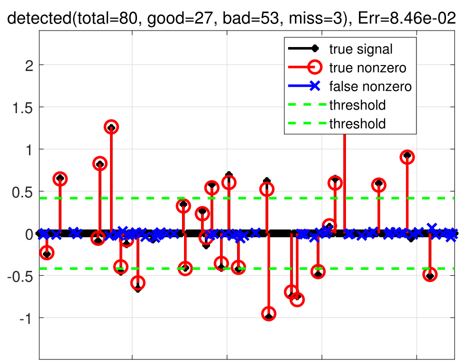
[1] We omit the 3rd-4th iterations in Figure 2 for consistency with Figure 1.



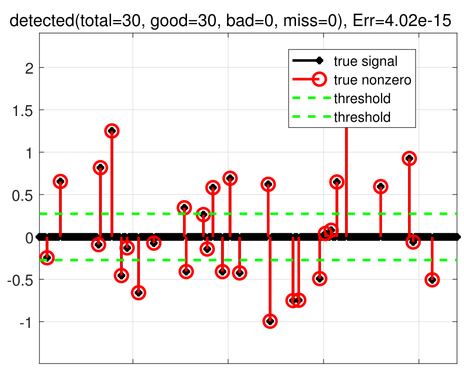
(a) The 1st iteration



(b) The 2nd iteration

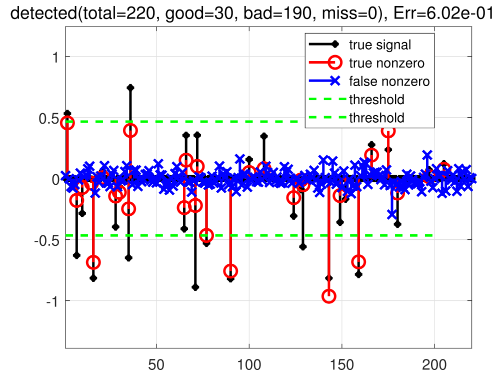


(c) The 3rd iteration

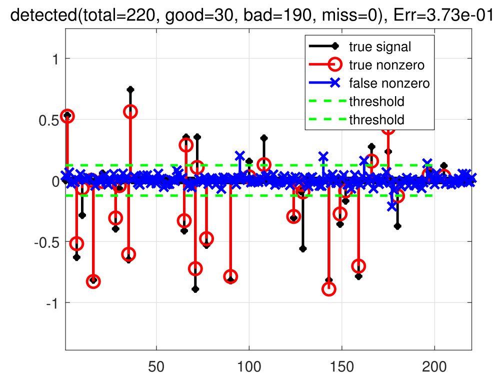


(d) The 4th iteration

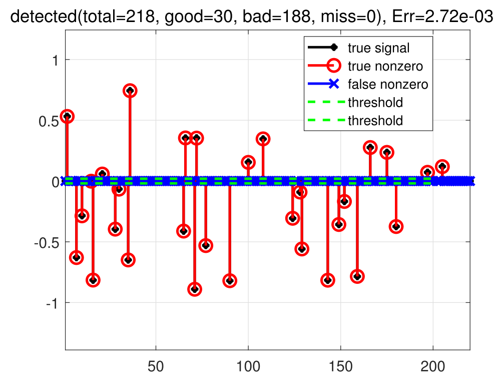
FIGURE 1. An ICDE-L1 demo that reconstructs a cosparse vector with $p = 220$, $d = 200$, $l = 190$, $m = 60$, $\beta = 0.5$. For the convenience of visualization, we create plots using the analyzed vectors Ωx instead of x.



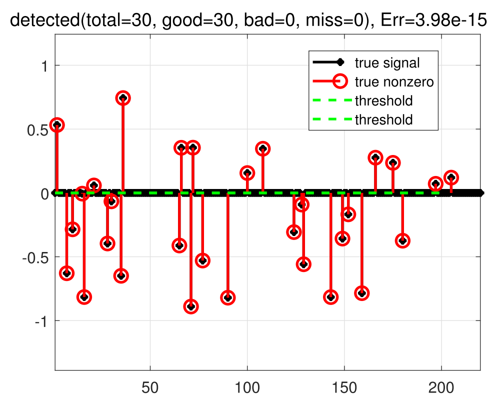
(a) The 1st iteration



(b) The 2nd iteration



(c) The 5th iteration



(d) The 6th iteration

FIGURE 2. An ICDE-L2 demo that reconstructs a cosparse vector with $p = 220$, $d = 200$, $l = 190$, $m = 80$, $\beta = 0.5$. For the convenience of visualization, we create plots using the analyzed vectors Ωx instead of x.

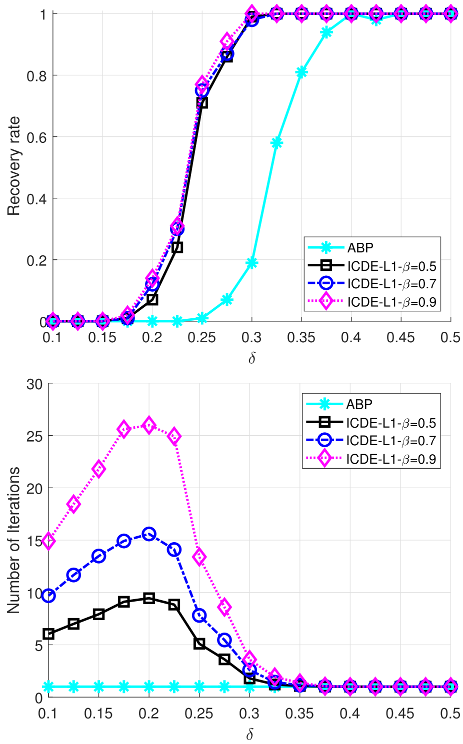


FIGURE 3. Influence of thresholding parameter β for ICDE-L1 recover cosparse signals with $p = 200$, $d = 200$, $l = 180$: comparisons in terms of recovery rate and number of iterations.

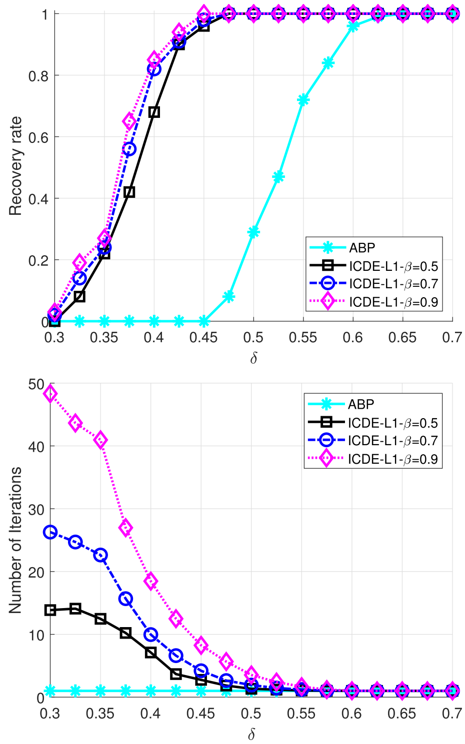


FIGURE 4. Influence of thresholding parameter β for ICDE-L1 recover cosparse signals with $p = 240$, $d = 200$, $l = 180$: comparisons in terms of recovery rate and number of iterations.

tion had exactly the same "good" coefficients as well as the "Err" almost as low as the double precision. Comparing with ICDE-L1 from Figure 1 and Figure 2, ICDE-L2 seems to resolve more easily but requires more iterations and measurements.

III. EXPERIMENTAL RESULTS

In this section, we carry out comparative experiments with GAP, ABP and ASP, which some of the experiments performed in [3], [29] for both synthetic and real-world datasets. GAP appears to be the state-of-the-art in terms of recovery performance. The code of GAP and ASP is available on the author's homepage. The super greedy version of GAP with the same thresholding parameter to ICDE-L2 was adopted hereafter. The truncated ABP problem (6) can be solved by a modification of MATLAB CVX package [43]. All the algorithmic implementations were tested in MATLAB 2018b running on Windows 7 with 2.8GHz Intel i7-7700HQ Quad Core CPU and 16GB of memory. A MATLAB package with code in this paper is available as open source software at https://github.com/songhp/ICDE.

We use standard independent and identically distributed Gaussian sensing matrix $M \in \mathbb{R}^{m\times d}$ and a random tight frame. The analysis operator $\Omega \in \mathbb{R}^{p\times d}$ was generated as a random almost-uniform almost-tight frame and Ω^T is a random tight-frame with normalized columns. We generate a l cosparse signal x in the following way. Firstly, randomly choose l indexes in $[1, p]$ denoted by Λ. Secondly, form a random vector v with Gaussian iid entries. Finally, project v onto the orthogonal complement of the subspace generated by Ω_Λ.

The signal dimension d is fixed to 200. We then varied the number of m measurements. The cosparsity l of the recovered signal, and the analysis operator size p are selected according to the following formulas:

$$m = \delta d, l = d - \rho m, p = \sigma d. \tag{12}$$

which is consistent with the notation of phase transition [37]: $\delta = m/d$ is the normalized measure of problem indeterminacy and $\rho = (d - l)/m$ is the normalized measure of the sparsity. In all cases, algorithm performance was quantified by recovery rate, i.e., the rate of perfect recovery on 100 times Monte Carlo problem instances. A relative error $\|\hat{x}^{(t)} - \hat{x}^{(t-1)}\|_2/\|\hat{x}^{(t)}\|_2$ less than 10^{-2} implies a perfect recovery of x.

A. INFLUENCE OF THRESHOLDING PARAMETER

We tested the influence of thresholding parameter β by varying β from 0.5 to 0.9. Firstly, we evaluated the ICDE-L1 algorithm. The plots of influence of thresholding parameter β for $\sigma = 1, 1.2$ are presented in Figure 3 and Figure 4, respectively. It is clear that the larger thresholding parameter exhibits better reconstruction performance but needs more iterations, especially for large σ. On the other hand, the improvement of recovery performance is limited to large β. In these cases, small β is particularly reasonable when the number of iterations is fairly small. We then turn to ABP, which was inferior to ICDE-L1. There has been

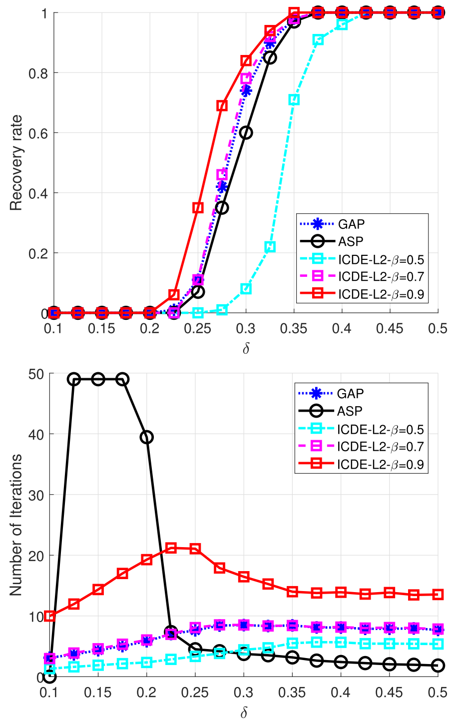


FIGURE 5. **Influence of thresholding parameter β for ICDE-L2 recover cosparse signals with $p = 200$, $d = 200$, $l = 180$: comparisons in terms of recovery rate and number of iterations.**

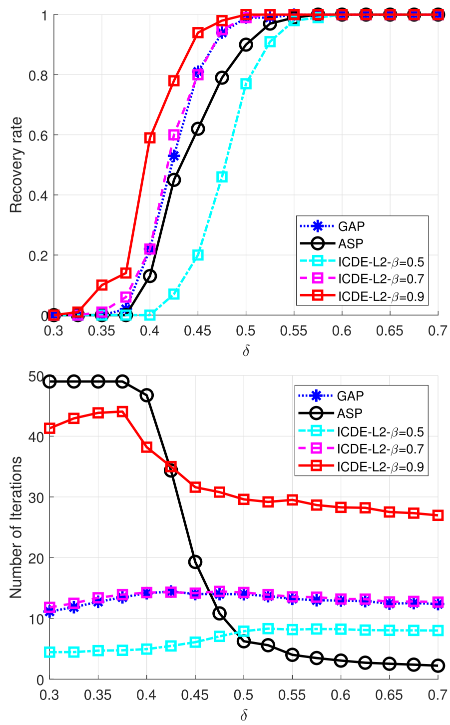


FIGURE 6. **Influence of thresholding parameter β for ICDE-L2 recover cosparse signals with $p = 240$, $d = 200$, $l = 180$: comparisons in terms of recovery rate and number of iterations.**

a great improvement in terms of recovery rate from ABP to ICDE-L1. Secondly, we tested the ICDE-L2 algorithm, as depicted in Figure 5 and Figure 6. The ICDE-L2 tests

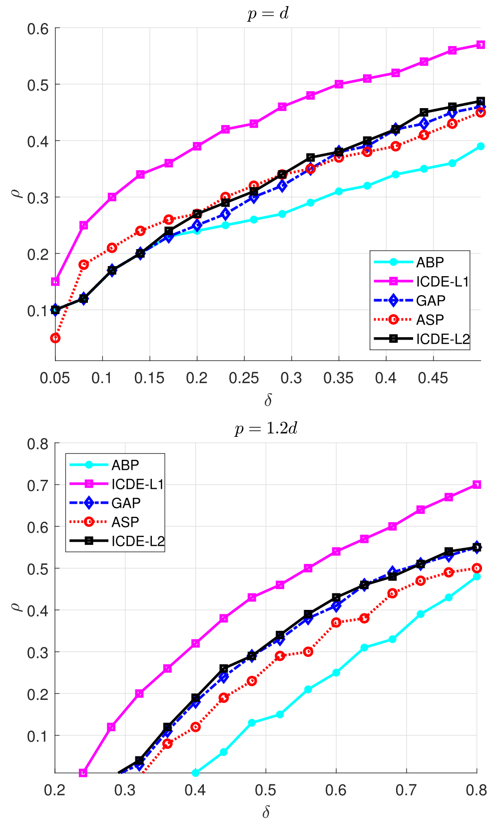


FIGURE 7. **Comparison of phase transitions of ABP, ASP, GAP and ICDE for cosparse vectors with $d = 200$, $\beta = 0.7$.**

came to similar conclusions to the ICDE-L1 tests. An increase in the parameter gave rise to an increase in performance in terms of recovery rate. An excessively small β leaded to too many false detections and subsequently low recovery quality, while an excessively large β could induce a large number of iterations. Large β is particularly reasonable when measurements size is fairly small. It is also worth noting that the iterating times are much smaller than the sparsity of Ωx, especially for large σ. We then turn to comparisons to GAP and ASP. ICDE-L2 was on par quality-wise with GAP and better than ASP for large β.

B. PHASE TRANSITION CURVE

In this subsection, we evaluated the recovery performance using the empirical phase transition curves (PTC) for ICDE, GAP, ASP and ABP. Figure 7 depicts the recovery performance of the five tested algorithms. For point in the grid (δ, ρ), it represents the probability of the cosparse signal recovered perfectly. For each problem sampling ratio $\delta = m/d$, we interpolate the results over all uncertainty ratios $\rho = (d - l)/m$ to locate where successful recovery takes place with a probability of 50%. As a result, we plot the PTC showing the boundary above which most recoveries fail and below which most recoveries succeed. Not surprisingly, the recoverability of the ABP method was the worst. In the case of $p = d$, ICDE-L2, GAP and ASP achieved comparable recoverability starting around $\delta = 0.18$, while ASP achieved

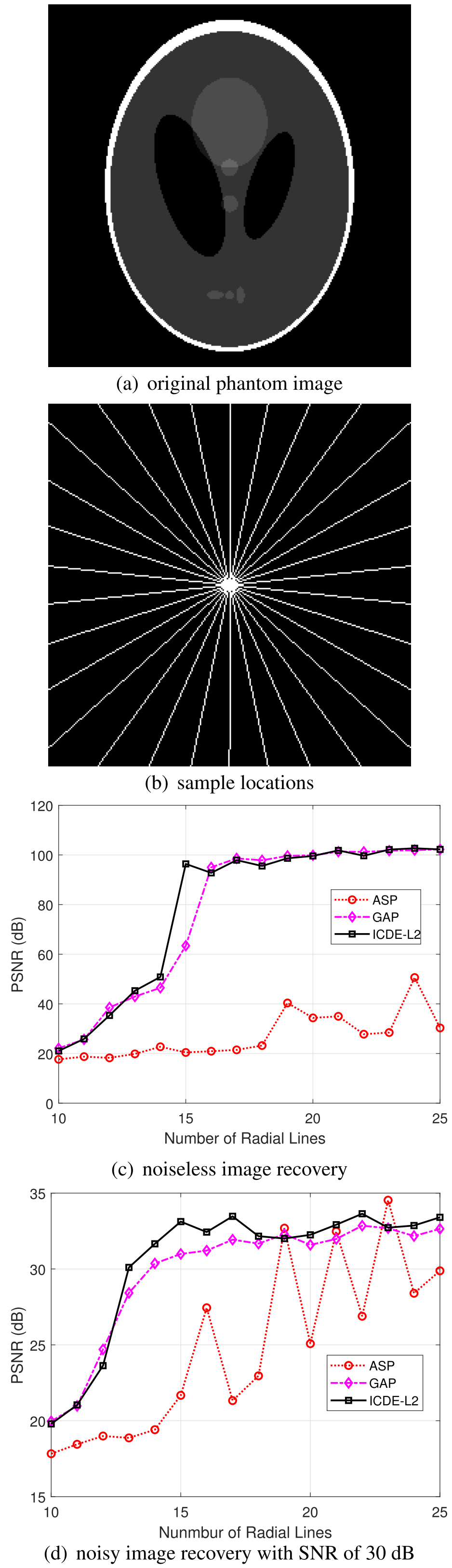


FIGURE 8. Shepp-Logan phantom reconstruction with size of 256 × 256: comparisons in terms of PSNR (dB).

better performance for some small δ cases. ICDE-L2 and GAP exhibited almost the same recoverability, which was much higher than that of ASP in terms of phase transition for the case $p = 1.2d$. ICDE-L1 uniformly obtained the best performance in the five tested algorithms and left significant gap between other algorithms. To sum up, Figure 7 depicts that these transitions comply with the following hierarchical sequence in recovery performance: ICDE-L1 > ICDE-L2 $\approx$ GAP > ASP > ABP.

C. ANALYSIS-BASED COMPRESSED SENSING

We considered the Shepp-Logan phantom image reconstruction problem as an ideal example that the cosparse analysis model offers effective algorithms to reconstruct partially observed cosparse signals. The image can only be indirectly observed by means of its two-dimensional Fourier transform coefficients that can only be directly observed along a small number of radial lines. We can recover the original image by the limited observations from two dimensional Fourier radial-line measurements $\mathrm{y} = M\mathrm{x}$, using an analysis operator Ω composed of two dimensional horizontal, vertical and diagonal finite-differences. To evaluate analysis-based compressed sensing, we performed comparison experiments by varying the number of radial observation lines for the 256×256 Shepp-Logan phantom image reconstruction problem. Due to the large size of the recovery problem, ICDE-L1 was time consuming and was omitted in these tests. Figure 8 depicts recovery peak SNR (PSNR) versus number of radial lines for ICDE-L2, GAP and ASP. The figure shows that ICDE-L2 achieved almost superior performance than GAP and ASP except for few radial lines, both in noiseless Figure 8(c) and noisy cases Figure 8(d). The original phantom image was presented in Figure 8(a) gave recovery PSNR approximately to 100 dB for all numbers of radial lines above 15. We obtained a perfect recovery using only 15 radial lines, i.e. only $m = 3782$ out of $d = 256 * 256$ which is less than 5.77% of the image size. The corresponding sampling locations were presented in Figure 8(b) using two-dimensional Fourier transform.

IV. CONCLUSIONS AND FUTURE WORK

In this paper, we generalized the Iterative Support Detection (ISD) and proposed an algorithmic framework, Iterative Cosupport Detection-Estimation (ICDE) for the sparse analysis recovery problem. ICDE can detect a cosupport set using the analyzed coefficients as a reference and then estimate the recovered signal by solving a truncated L1 or L2 optimization problem on the cosupport set, and repeat these two iterations for a small number of iterations. We proposed two implementations of ICDE, accompanying a cosupport detection strategy. Numerical experiments on both synthetic and real-world datasets show that the proposed methods compare favorably with the state-of-the-art methods. The encouraging performance of the proposed methods prompts us to pursue further studies: 1).This paper dedicates efforts to devising numerical algorithms which are highly reproducible and provide experimental studies as constructive guidelines for practical applications, further, the theoretical investigation of ICDE can be developed for future research. 2). ICDE requires reliable cosupport detection from inexact recovery. We present a thresholding strategy with not necessarily decreasing or nested cosupport. This method exhibits efficient performance for the cases in which the analyzed coefficients follow a fast-decaying distribution of non-zeros. Therefore future investigations can take advantage of the informative prior about the true signal to devise more effective cosupport detection methods. 3). In signal estimation stage, we employ the truncated L1 or L2 optimization problem to update the reconstructed signal. ICDE-L1 performed significantly better than ICDE-L2. Since ABP is not the only algorithm for cosparse signal recovery, another line of future investigation is to apply ICDE to other reconstruction algorithms such as the smoothing-based accelerated alternating minimization [31], reweighted approaches [23], [32], matching pursuit generalized LASSO [33], sophisticated cosparsity inducing function [24], [44], [45], and many others.

REFERENCES

[1] R. G. Baraniuk, E. Candes, M. Elad, and Y. Ma, "Applications of sparse representation and compressive sensing," *Proc. IEEE*, vol. 98, no. 6, pp. 906–909, May 2010.

[2] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. New York, NY, USA: Springer, 2010.

[3] S. Nam, M. E. Davies, M. Elad, and R. Gribonval, "The cosparse analysis model and algorithms," *Appl. Comput. Harmon. Anal.*, vol. 34, no. 1, pp. 30–56, Jan. 2013.

[4] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Rev.*, vol. 51, no. 1, pp. 34–81, Feb. 2009.

[5] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proc. IEEE*, vol. 98, no. 6, pp. 1031–1044, Jun. 2010.

[6] J. Mairal, F. Bach, and J. Ponce, "Sparse modeling for image and vision processing," *Found. Trends Comput. Graph. Vis.*, vol. 8, nos. 2–3, pp. 283–285, 2014.

[7] Z. Zhang, Y. Xu, J. Yang, X. Li, and D. Zhang, "A survey of sparse representation: Algorithms and applications," *IEEE Access*, vol. 3, pp. 490–530, 2015.

[8] Y. Wang, D. Meng, and M. Yuan, "Sparse recovery: From vectors to tensors," *Nat. Sci. Rev.*, vol. 5, no. 5, pp. 756–767, Sep. 2018.

[9] E. J. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[10] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[11] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, Jan. 2001.

[12] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. 27th Asilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, USA, vol. 1, Nov. 1993, pp. 40–44.

[13] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[14] 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.

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

[16] S. Foucart, "Hard thresholding pursuit: An algorithm for compressive sensing," *SIAM J. Numer. Anal.*, vol. 49, no. 6, pp. 2543–2563, Jan. 2011.

[17] J. Huang, Y. Jiao, Y. Liu, and X. Lu, "A constructive approach to l_0 penalized regression," *J. Mach. Learn. Res.*, vol. 19, no. 10, pp. 1–37, 2018.

[18] M. Elad, P. Milanfar, and R. Rubinstein, "Analysis versus synthesis in signal priors," *Inverse Problems*, vol. 23, no. 3, pp. 947–968, Jun. 2007.

[19] S. Vaiter, G. Peyre, C. Dossal, and J. Fadili, "Robust sparse analysis regularization," *IEEE Trans. Inf. Theory*, vol. 59, no. 4, pp. 2001–2016, Apr. 2013.

[20] M. Genzel, G. Kutyniok, and M. Marz, "ℓ_1-analysis minimization and generalized (co-)sparsity: When does recovery succeed?" *Appl. Comput. Harmon. Anal.*, vol. 52, pp. 82–140, May 2021.

[21] R. J. Tibshirani and J. Taylor, "The solution path of the generalized lasso," *Ann. Statist.*, vol. 39, no. 3, pp. 1335–1371, Jun. 2011.

[22] S. Nam, M. E. Davies, M. Elad, and R. Gribonval, "Recovery of cosparse signals with greedy analysis pursuit in the presence of noise," in *Proc. 4th IEEE Int. Workshop Comput. Adv. Multi-Sensor Adapt. Process. (CAMSAP)*, Dec. 2011, pp. 361–364.

[23] Z. Liu, J. Li, W. Li, and P. Dai, "A modified greedy analysis pursuit algorithm for the cosparse analysis model," *Numer. Algorithms*, vol. 74, no. 3, pp. 867–887, Mar. 2017.

[24] F. Mohagheghian, M. R. Deevband, N. Samadzadehaghdam, H. Khajehpour, and B. Makkiabadi, "An enhanced weighted greedy analysis pursuit algorithm with application to EEG signal reconstruction," *Int. J. Imag. Syst. Technol.*, vol. 30, no. 4, pp. 1–13, 2020.

[25] R. E. Carrillo, K. E. Barner, and T. C. Aysal, "Robust sampling and reconstruction methods for sparse signals in the presence of impulsive noise," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 392–408, Apr. 2010.

[26] D. Wu, Y. Zhao, W. Wang, and Y. Hao, "Cosparsity-based stagewise matching pursuit algorithm for reconstruction of the cosparse signals," *EURASIP J. Adv. Signal Process.*, vol. 2015, no. 101, pp. 1–11, Dec. 2015.

[27] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck, "Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 58, no. 2, pp. 1094–1121, Feb. 2012.

[28] R. Giryes and M. Elad, "CoSaMP and SP for the cosparse analysis model," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2012, pp. 964–968.

[29] R. Giryes, S. Nam, M. Elad, R. Gribonval, and M. E. Davies, "Greedy-like algorithms for the cosparse analysis model," *Linear Algebra Appl.*, vol. 441, pp. 22–60, Jan. 2014.

[30] M. Borgerding, P. Schniter, J. Vila, and S. Rangan, "Generalized approximate message passing for cosparse analysis compressive sensing," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, South Brisbane, QLD, Australia, Apr. 2015, pp. 3756–3760.

[31] J. Xie, A. Liao, and Y. Lei, "A new accelerated alternating minimization method for analysis sparse recovery," *Signal Process.*, vol. 145, pp. 167–174, Apr. 2018.

[32] R. Ahmad and P. Schniter, "Iteratively reweighted approaches to sparse composite regularization," *IEEE Trans. Comput. Imag.*, vol. 1, no. 4, pp. 220–235, Dec. 2015.

[33] D. Gong, M. Tan, Y. Zhang, A. van den Hengel, and Q. Shi, "MPGL: An efficient matching pursuit method for generalized LASSO," in *Proc. 31st AAAI Conf. Artif. Intell.*, San Francisco, CA, USA, Feb. 2017, pp. 1934–1940.

[34] S. Boyd, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2010.

[35] Y. Wang and W. Yin, "Sparse signal reconstruction via iterative support detection," *SIAM J. Imag. Sci.*, vol. 3, no. 3, pp. 462–491, Jan. 2010.

[36] A. A. Amini, M. Babaie-Zadeh, and C. Jutten, "A fast method for sparse component analysis based on iterative detection-estimation," *AIP Conf.*, vol. 872, no. 1, pp. 123–130, 2006.

[37] A. Maleki and D. L. Donoho, "Optimally tuned iterative reconstruction algorithms for compressed sensing," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 330–341, Apr. 2010.

[38] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Appl. Comput. Harmon. Anal.*, vol. 27, no. 3, pp. 265–274, Nov. 2009.

[39] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, Jan. 2009.

[40] I. Bayram and I. W. Selesnick, "A subband adaptive iterative shrinkage/thresholding algorithm," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1131–1143, Mar. 2010.

[41] H. Song, Z. Ai, Y. Lai, H. Meng, and Q. Mao, "Sparse signal reconstruction via generalized two-stage thresholding," *Sci. China Inf. Sci.*, vol. 64, no. 5, pp. 1–2, 2021.

[42] H. Song and G. Wang, "Sparse signal recovery via ECME thresholding pursuits," *Math. Problems Eng.*, vol. 2012, pp. 1–22, May 2012.
[43] M. Grant and S. Boyd. (2014). *CVX: MATLAB Software for Disciplined Convex Programming Version 2.1*. [Online]. Available: https://cvxr.com/cvx
[44] G. Wang, H. Zhang, S. Yu, and S. Ding, "A family of the subgradient algorithm with several cosparsity inducing functions to the cosparse recovery problem," *Pattern Recognit. Lett.*, vol. 80, pp. 64–69, Sep. 2016.
[45] Z. Li, S. Ding, Y. Li, Z. Yang, S. Xie, and W. Chen, "Manifold optimization-based analysis dictionary learning with an $\ell_{1/2}$-norm regularizer," *Neural Netw.*, vol. 98, pp. 212–222, Feb. 2018.

HEPING SONG received the Ph.D. degree in computer application technology from Sun Yat-sen University, Guangzhou, China, in 2011. He is currently an Associate Professor with the Department of Software Engineering, School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, China. His current research interests include low-level vision, inverse problems, computer vision, and deep learning.

XIAOLONG REN received the B.S. degree in software engineering from the School of Computer Science and Communication Engineering, Jiangsu University, China, in 2018, where he is currently pursuing the M.S. degree. His research interests include image processing and sparse representation.

YUPING LAI received the Ph.D. degree in information security from the Beijing University of Posts and Telecommunications, Beijing, China, in 2014. He is currently an Associate professor with the Beijing University of Posts and Telecommunications. His research interests include information security, computer vision, pattern recognition, machine learning, and data mining.

HONGYING MENG (Senior Member, IEEE) received the Ph.D. degree in communication and electronic systems from Xi'an Jiaotong University. He was a Lecturer with the Department of Electronic Engineering, Tsinghua University, China. He is currently a Reader with the Department of Electronic and Electrical Engineering, Brunel University London, U.K. His research interests include digital signal processing, machine learning, human–computer interaction, computer vision, image processing, and embedded systems.