Fast Signal Separation of 2D Sparse Mixture via Approximate Message-Passing

Jaewook Kang, *Student Member, IEEE,* Hyoyoung Jung, *Student Member, IEEE,*
and Kiseon Kim, *Senior Member, IEEE*

Abstract—Approximate message-passing (AMP) method is a simple and efficient framework for the linear inverse problems. In this letter, we propose a faster AMP to solve the L_1-*Split-Analysis* for the 2D sparsity separation, which is referred to as *MixAMP*. We develop the MixAMP based on the factor graphical modeling and the min-sum message-passing. Then, we examine MixAMP for two types of the sparsity separation: separation of the direct-and-group sparsity, and that of the direct-and-finite-difference sparsity. This case study shows that the MixAMP method offers computational advantages over the conventional first-order method, TFOCS.

Index Terms—2D Compressed sensing, Sparse signal separation, approximate message-passing (AMP), L_1-Split-Analysis.

I. INTRODUCTION

The linear inverse problems for estimating an unknown signal $\mathbf{X}$ from linear measurements $\mathbf{Y}$ are central in signal processing techniques. In this letter, we revisit such inverse problems with three additives:

- $\mathbf{X}$ is a two-dimensional (2D) square, whose entries are with the index pair $(i, j) \in \Lambda := \{1, ..., \sqrt{N}\} \times \{1, ..., \sqrt{N}\}$ where $\sqrt{N} \in \mathbb{Z}^+$ and $N := |\Lambda|$,
- $\mathbf{X}$ is a mixture of two signals $\mathbf{X}_a, \mathbf{X}_b$ having sparse representation in two separate basis, *i.e.*, $\mathbf{X} = \mathbf{X}_a + \mathbf{X}_b$.
- $\mathbf{Y}$ are *Compressed Sensing* (CS) measurements such that $\mathbf{Y}$ includes its effective samples with the index pair $(k, l) \in \Omega \subseteq \Lambda$ where $M := |\Omega|$.

From the 2D setup, we generate the linear measurements by

$$\mathbf{Y} = \mathcal{P}_\Omega \left\{ \mathbf{A}\mathbf{X}\mathbf{A}^T \right\} \in \mathbb{R}^{\sqrt{N} \times \sqrt{N}}, \qquad (1)$$

where $\mathbf{A} \in \mathbb{R}^{\sqrt{N} \times \sqrt{N}}$ is a measurement matrix; $\mathcal{P}_\Omega\{\cdot\}$ is the undersampling operator nulling entries of its matrix argument not in the set Ω. The measurement model (1) is motivated by the 2D signal acquisition concept considering two perpendicular spatial/spectral axes of $\mathbf{X}$ where "the left $\mathbf{A}$" linearly mixes $\mathbf{X}$ in vertical axis and "the right $\mathbf{A}^T$" does in horizontal axis. As practical applications of (1), the 2D Fourier-transform-based image/video acquisition [8],[15] and optical image encryption [1] have been considered.

The authors are with Department of Information and Communication, Gwangju Institute of Science and Technology, Gwangju, Republic of Korea (Email:{jwkkang,rain,kskim}@gist.ac.kr)

The measurement model (1) has two advantages in 2D signal processing. This 2D model spends $\mathcal{O}(N^{3/2})$ complexity for the measurement generation, which is less than $\mathcal{O}(N^2)$ by the 1D equivalent model [2],

$$\text{vec}(\mathbf{Y}) = \mathcal{P}_{\Omega'} \left\{ (\mathbf{A} \otimes \mathbf{A}) \, \text{vec}(\mathbf{X}) \right\} \in \mathbb{R}^{N \times 1}, \qquad (2)$$

where $\otimes$ is Kronecker product, $\text{vec}(\cdot)$ is the columnwise vectorizing operator, and $\Omega' \subseteq \{1, ..., N\}$. In addition, the 2D model is storage efficient by saving memory for the Kronecker product matrix $\mathbf{A} \otimes \mathbf{A} \in \mathbb{R}^{N \times N}$ which is unavoidable under the 1D model (2).

Our main task is to simultaneously estimate the two 2D signals, $\mathbf{X}_a, \mathbf{X}_b$, given the knowledge of $\mathbf{Y}$, $\mathbf{A}$, and Ω under (1). This problem so-called *Sparse signal separation*, being related to the *Analysis CS* whose unknown signal is sparse in a concatenation of the two basis [3], including applications to image inpainting and deblurring [5], and super-resolution [6].

When the cardinality of the sets has $|\Omega| < |\Lambda|$ such that the problems are ill-posed, optimization methods with regularization have been mostly considered. A known approach to the problem is the L_1-*Split-Analysis* [3]-[5]:

$$\begin{cases} \min_{\mathbf{X}_a, \mathbf{X}_b} & ||\mathcal{T}_a \mathbf{X}_a||_1 + ||\mathcal{T}_b \mathbf{X}_b||_1 \\ \text{s.t.} & \mathbf{Y} = \mathcal{P}_\Omega \left\{ \mathbf{A}(\mathbf{X}_a + \mathbf{X}_b)\mathbf{A}^T \right\}, \end{cases} \qquad (3)$$

where the sparsity of $\mathbf{X}_a, \mathbf{X}_b$ is promoted in terms of analysis transform operators $\mathcal{T}_a, \mathcal{T}_b$. This L_1-Split-Analysis is related to *Morphological Component Analysis* (MCA) [13],[14] in that the both analysis approaches decompose $\mathbf{X}$ by pursuing sparse representation of $\mathbf{X}_a, \mathbf{X}_b$. However, they are distinct in that MCA does not include measurement compression such as (1) but simply goes with $\mathbf{Y} = \mathbf{X}_a + \mathbf{X}_b$.

Practical solving of (3) has been considered in the works of [4],[5] via the *Templates for First-Order Conic Solvers* (TFOCS) [7] and the *Split Bregman iteration* [8], respectively. However, their examples are not in the context of our problem setup since they do not contain the concept of linear mixing by setting $\mathbf{A} = \mathbf{I}$.

In this letter, we propose an *Approximate Message-Passing* (AMP) method for solving the L_1-*Split-Analysis* under the 2D CS model (1). This is motivated by excellent properties of AMP [9],[10]: *i)* asymptotic Lasso performance, *ii)* efficient computations, and *iii)* algorithmic simplicity. We refer to the proposed method as *separation of sparse mixture via approximate message-passing* (MixAMP). We claim that MixAMP is remarkably faster than the conventional first-order method, TFOCS [7], for the 2D sparse signal separation task.

We believe that another advantage of this MixAMP lies in its flexibility. The 2D separation problems, regarding various types of the sparsity, can be solved via the MixAMP once

Fig. 1. Factor graphical model and message-passing for the L_1-Split-Analysis problem (3).

Algorithm 1 MixAMP Method

Require: Measurement matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$, Measurements $\mathbf{Y} \in \mathbb{R}^{N \times N}$, Undersampling operator $\mathcal{P}_\Omega\{\cdot\}$
Ensure: Recovered signals $\mathbf{X}_a^{t=t^*}, \mathbf{X}_b^{t=t^*} \in \mathbb{R}^{N \times N}$

Initialization: $\mathbf{R}^{t=0} = \mathbf{Y}, \theta^{t=0} = \frac{1}{M}\|\mathbf{Y}\|_F^2$
$\qquad\qquad \mathbf{X}_a^{t=0} = \mathbf{0}, \mathbf{X}_b^{t=0} = \mathbf{0}$

for $t = 1$ **to** t^* do **do**
$\quad \mathbf{X}_a^t = \eta_a(\mathbf{A}^T\mathbf{R}^{t-1}\mathbf{A} + \mathbf{X}_a^{t-1}; \theta^{t-1})$
$\quad \mathbf{X}_b^t = \eta_b(\mathbf{A}^T\mathbf{R}^{t-1}\mathbf{A} + \mathbf{X}_b^{t-1}; \theta^{t-1})$
$\quad \mathbf{R}^t = \mathbf{Y} - \mathcal{P}_\Omega\{\mathbf{A}(\mathbf{X}_a^t + \mathbf{X}_b^t)\mathbf{A}^T\}$
$\qquad\qquad + \mathbf{R}^{t-1} \frac{N}{M} \left\langle \eta_a{}'(\mathbf{A}^T\mathbf{R}^{t-1}\mathbf{A} + \mathbf{X}_a^{t-1}; \theta^{t-1}) \right\rangle$
$\qquad\qquad + \mathbf{R}^{t-1} \frac{N}{M} \left\langle \eta_b{}'(\mathbf{A}^T\mathbf{R}^{t-1}\mathbf{A} + \mathbf{X}_b^{t-1}; \theta^{t-1}) \right\rangle$
$\quad \theta^t = \frac{1}{M}\|\mathbf{R}^t\|_F^2$
end for

---

a proper denoiser is given. In the sequel, we examine the MixAMP for two cases of the sparsity separation, *i)* separation of direct-and-group sparse mixture, and *ii)* that of direct-and-finite-difference (FD) sparse mixture, by applying the simple denoisers introduced in the literature [9],[10]. In each case, we demonstrate the low-computationality of MixAMP by an exemplary comparison to the TFOCS method.[1]

II. AMP METHOD FOR L_1-SPLIT-ANALYSIS

The first step of the AMP development is to establish a message-passing rule based on a factor graphical model of linear systems. Fig.1 describes a factor graph for our system (1), deriving the corresponding joint PDF, given by

$$f_{\mathbf{X}_a,\mathbf{X}_b,\mathbf{Y}}(\cdot) \propto f_{\mathbf{X}_a}(\cdot;\mathcal{T}_a)f_{\mathbf{X}_b}(\cdot;\mathcal{T}_b) \prod_{(k,l)\in\Omega} f_{Y_{kl}|\mathbf{X}_a,\mathbf{X}_b}(\cdot) \quad (4)$$

where the two priors, $f_{\mathbf{X}_a}(\cdot;\mathcal{T}_a)$ and $f_{\mathbf{X}_b}(\cdot;\mathcal{T}_b)$, sparsify the signals in the domain of $\mathcal{T}_a$ and $\mathcal{T}_b$, respectively. From the joint PDF (4), we form a message-passing rule for solving (3) in the manner of the *min-sum algorithm* [11], given by

a) For the signal $\mathbf{X}_a : \forall(ji, kl) \in \Lambda \times \Omega$

$$\begin{cases} \mu_{kl\to ji}(x_{a,ji}) = \min_{\substack{\{\mathbf{x}_a,\mathbf{x}_b\} \\ \backslash x_{a,ji}}} \log f_{Y_{kl}|\mathbf{X}_a,\mathbf{X}_b}(\cdot) + \sum_{\substack{j'i' \\ \neq ji}} \alpha_{j'i'\to kl}(x_{a,j'i'}) \\ \qquad\qquad\qquad + \sum_{ji\in\Lambda} \beta_{ji\to kl}(x_{b,ji}) \\ \alpha_{ji\to kl}(x_{a,ji}) = \log f_{\mathbf{X}_a}(\mathbf{x}_a;\mathcal{T}_a) + \sum_{k'l'\neq kl} \mu_{k'l'\to ji}(x_{a,ji}) \end{cases}$$

b) For the signal $\mathbf{X}_b : \forall(ji, kl) \in \Lambda \times \Omega$

$$\begin{cases} \nu_{kl\to ji}(x_{b,ji}) = \min_{\substack{\{\mathbf{x}_a,\mathbf{x}_b\} \\ \backslash x_{b,ji}}} \log f_{Y_{kl}|\mathbf{X}_a,\mathbf{X}_b}(\cdot) + \sum_{ji\in\Lambda} \alpha_{ji\to kl}(x_{b,ji}) \\ \qquad\qquad\qquad + \sum_{j'i'\neq ji} \beta_{j'i'\to kl}(x_{b,j'i'}) \\ \beta_{ji\to kl}(x_{b,ji}) = \log f_{\mathbf{X}_b}(\mathbf{x}_b;\mathcal{T}_b) + \sum_{k'l'\neq kl} \nu_{k'l'\to ji}(x_{b,ji}), \end{cases}$$

$$(5)$$

---

[1]The MATLAB codes for this comparison is available at our website, https://sites.google.com/site/jwkang10/.

---

where $\{\mathbf{x}\}$ indicates the set of elements in the vector argument $\mathbf{x}$. As illustrated in Fig.1, the min-sum rule (5) disjointly exchanges the messages between onefactor layer and two different variable layers where the factor nodes are effective only with the index $(k,l) \in \Omega$, and the variable nodes correspond to the index $(i,j) \in \Lambda$. Then, totally, $4MN$ messages are handled per iteration since there are M effective factors, each of which generates $2N$ messages, and $2N$ variables, each of which produces M messages.

At the fixed-point, the scalar MAP estimate of each signal is approximated by

$$\widehat{x}_{a,ji} = \arg \min_{x_{a,ji}\in\mathbb{R}} \alpha_{ji}(x_{a,ji}), \quad \widehat{x}_{b,ji} = \arg \min_{x_{b,ji}\in\mathbb{R}} \beta_{ji}(x_{b,ji}), \quad (6)$$

where the functions $\alpha_{ji}(x_{a,ji}), \beta_{ji}(x_{b,ji})$, which include the posterior information, are given by

$$\alpha_{ji}(x_{a,ji}) = \log f_{\mathbf{X}_a}(\mathbf{x}_a;\mathcal{T}_a) + \sum_{(k,l)\in\Omega} \mu_{kl\to ji}(x_{a,ji}), \quad (7)$$

$$\beta_{ji}(x_{b,ji}) = \log f_{\mathbf{X}_b}(\mathbf{x}_b;\mathcal{T}_b) + \sum_{(k,l)\in\Omega} \nu_{kl\to ji}(x_{b,ji}). \quad (8)$$

In the AMP literature [9]-[11], the second term of (7),(8) are handled as Gaussian exponents. This is based on the assumption that the factor graph connection, by the matrix $\mathbf{A}$, is sufficiently dense such that the messages $\mu_{kl\to ji}(x_{a,ji}), \nu_{kl\to ji}(x_{b,ji})$ from each factor are Gaussian distributed by the law of large numbers.

The remaining steps for the AMP development consist of

1) *the quadratic approximation step*, which approximates the min-sum equations by quadratic functions, then converting (5) to a parameter-passing rule whose messages are simple real numbers (instead of functions),
2) *the first-order approximation step*, which cancels interference caused by the loopy graph connection, and reduces the number of messages from $4MN$ to $M+2N$.

These development steps are conventional for the AMP methods applied to the other types of linear inverse problems [9]-[11], which are well formulated in the literature [11]. Therefore, we omit details of such remaining steps in this letter, immediately providing a final form of MixAMP in Algorithm 1.

One thing noteworthy is that in Algorithm 1 the two disjoint iterations, described in Fig.1, share the residual term $\mathbf{R}^t \in \mathbb{R}^{N \times N}$, which significantly reduces the number of the messages in the iteration. This can be explained using the two arguments given in [11]: *i*) The messages, sent by the (k,l)-th factor, have a residual form through the quadratic approximation step; for example, the message toward the variable index (a, ji) is expressed as

$$r_{a,kl \to ji} = y_{kl} - \mathcal{P}_\Omega \Big\{ \sum_{j'i' \neq ji} a_{i'j'} a_{j'i'} x_{a,j'i' \to kl}$$
$$+ \sum_{ji \in \Lambda} a_{ij} a_{ji} x_{b,ji \to kl} \Big\}. \quad (9)$$

ii) We can drop the directional dependency upon the destination index (a, ji) in (9) by decomposing the residual message $r_{a,kl \to ji}$ into a form of "pure residual + directional correction" under the large limit $M, N \to \infty$, *i.e.*,

$$r_{a,kl \to ji} = r_{kl} + \delta r_{a,kl \to ji}$$
$$= \underbrace{y_{kl} - \mathcal{P}_\Omega \Big\{ \sum_{ji \in \Lambda} a_{ij} a_{ji} (x_{a,ji \to kl} + x_{b,ji \to kl}) \Big\}}_{=r_{kl}}$$
$$+ \underbrace{a_{ij} a_{ji} x_{a,ji}}_{=\delta r_{a,kl \to ji}}, \quad (10)$$

where the correction term $\delta r_{a,kl \to ji}$ has the order of $\mathcal{O}(N^{-1/2})$. The above two arguments equivalently hold for the factor messages toward the index (b, ji). Therefore, the pure residual $r_{kl}^t \in \mathbf{R}^t$ is independent of the index of the destination variable, enabling the residual sharing in the two disjoint iteration.

The MixAMP incorporates two distinct denoisers, denoted by $\eta_a(\cdot), \eta_b(\cdot)$, according to sparsity types in the mixture $\mathbf{X}_a + \mathbf{X}_b$. These denoisers undertake the sub-optimization tasks given in (6), generating the MAP estimate $\mathbf{X}_a^t, \mathbf{X}_b^t$ at every iteration. Hence, choice of the sparsifying priors, $f_{\mathbf{X}_a}(\cdot), f_{\mathbf{X}_b}(\cdot)$, determines functional form of the denoisers; for some choices, we may need to utilize external numerical solvers for the denoiser implementation due to analytical difficulties of the prior exponent *e.g.*, non-scalability and non-smoothness. This denoising concept of MixAMP is analogous to the shrinkage in the context of *iterative shrinkage-thresholding* (IST). However, they are different in that the MixAMP denoisers shrink $\mathbf{X}_a^t, \mathbf{X}_b^t$ in the domain of $\mathcal{T}_a$ and $\mathcal{T}_b$ respectively, whereas the shrinkage operator of IST takes soft-thresholding in the standard domain.

III. CASE STUDIES FOR 2D SPARSITY SEPARATION

This section provides an exemplary comparison of the MixAMP and the TFOCS [7] methods for two different cases of the sparsity separation. In each case, we first introduce denoisers applied to the MixAMP method, and then we provide the separation example. We inform that the TFOCS method is not directly applicable to the 2D model (1), which should be accompanied by the vectorization of (2). This comparison is based on the standard Gaussian matrix $\mathbf{A}$ whose entries are *i.i.d.* drawn from $\mathcal{N}(0, \frac{1}{M})$, and MATLAB 2014a with a 2.67-GHz Intel Quad Core *i5* was used to generate the results. In addition, we stop the MixAMP and TFOCS iteration when $tol := \frac{\sqrt{||\mathbf{X}_a^{t-1} - \mathbf{X}_a^t||_F^2 + ||\mathbf{X}_b^{t-1} - \mathbf{X}_b^t||_F^2}}{\sqrt{||\mathbf{X}_a^t||_F^2 + ||\mathbf{X}_b^t||_F^2}} \leq 5 \times 10^{-4}$ is met.

70% Sampling for Y 5% sparsity of Xa $tol = 5 \times 10^{-4}$		
Solver Type	PSNR of Xb <dB>	Runtime <sec>
MixAMP	26.83	2.02
TFOCS	26.69	74.20

Fig. 2. An exemplary comparison of MixAMP and TFOCS in the direct-and-group sparsity separation where we reconstruct a 128×128 QRcode image $\mathbf{X}_b$ by removing shot noise $\mathbf{X}_a$ (5% sparsity) given the measurements $\mathbf{Y}$ with 70% sampling ($M/N = 0.7$). Here, we set $\varepsilon = 10^{-10}$, $\lambda_1 = 0.5$, $\lambda_2 = 1.2$ for the TFOCS method (13).

A. Separation of Direct and Group Sparsity

We consider separation of a 2D direct-and-group sparse mixture. For the direct sparsity denoiser, we apply the *soft-thresholding* which has been the most widely used because of its simplicity [9]. Let $\mathbf{X}_a \in \mathbb{R}^{\sqrt{N} \times \sqrt{N}}$ have the direct sparsity. Then, we can estimate $X_{a,ji} \in \mathbb{R}$ via a scalable denoiser $\eta_a(\cdot) : \mathbb{R} \to \mathbb{R}$, given by

$$\widehat{x}_{a,ji} = \eta_a(x; \theta) := \arg \min_{X_{a,ji}} |X_{a,ji}| + \frac{\lambda}{2} (X_{a,ji} - x)^2$$
$$= \text{sgn}(x) \max\{|x| - \theta, 0\}. \quad (11)$$

In (11), the input $x \in \mathbb{R}$ is a Gaussian variable corresponding to the second term of the posterior function (7). This Gaussianity let the denoiser to solve a penalized least squares problem, which is common in the AMP denoisers. In addition, it is well known that the soft-thresholding is based on the Laplace prior; therefore, we have $\log f_{\mathbf{X}_a}(\mathbf{x}_a) = ||\mathbf{X}_a||_1$ in (7).

We adopt the *block soft-thresholding* for the group sparsity pursuit [10]. Let $\mathbf{X}_b \in \mathbb{R}^{\sqrt{N} \times \sqrt{N}}$ have the group

sparsity. Then, the block soft-thresholding denoiser, $\eta_b(\cdot) : \mathbb{R}^{\sqrt{B}\times\sqrt{B}} \to \mathbb{R}^{\sqrt{B}\times\sqrt{B}}$, is given by

$$\widehat{\mathbf{x}}_{b,B} = \eta_b(\mathbf{x}_B; \theta) := \arg\min_{\mathbf{X}_{b,B}} ||\mathbf{X}_{b,B}||_F + \frac{\lambda}{2}||\mathbf{X}_{b,B} - \mathbf{x}_B||_F^2$$

$$= \mathbf{x}_B \cdot \max\{1 - \theta||\mathbf{x}_B||_F^{-1}, 0\} \qquad (12)$$

where the input $\mathbf{x}_B \in \mathbb{R}^{\sqrt{B}\times\sqrt{B}}$ is a block matrix such that the signal $\mathbf{X}_b$ is partitioned into N/B square blocks with the size B. Specifically, this block denoiser makes its block argument $\mathbf{x}_B$ to a zero matrix if $||\mathbf{x}_B||_F \leq \theta$, otherwise diminishing it by the quantity θ to the origin. In addition, this block thresholding is related to a block Gaussian prior; hence, we have $\log f_{\mathbf{X}_b}(\mathbf{x}_b) = \sum_{\text{All blocks}} ||\mathbf{X}_{b,B}||_F$ from (8).

Fig.2 displays a separation example by MixAMP and TFOCS, where we reconstruct a 128×128 QRcode image $\mathbf{X}_b$ by removing shot noise $\mathbf{X}_a$ given the measurements $\mathbf{Y}$ with 70% sampling ($M/N = 0.7$). The QR code refers to the group sparsity part of the mixture image, while the shot noise models the direct sparsity part. In the TFOCS method, we recast (3) to *Combining-L_1-and-Group minimization*, solving

$$\min \quad \lambda_1||\mathbf{X}_a||_1 + \lambda_2 \sum_{\text{All blocks}} ||\mathbf{X}_{b,B}||_F$$
$$\text{s.t.} \quad ||\mathbf{Y} - \mathcal{P}_\Omega\{\mathbf{A}(\mathbf{X}_a + \mathbf{X}_b)\mathbf{A}^T\}||_F \leq \varepsilon \qquad (13)$$

where $\varepsilon, \lambda_1, \lambda_2 \geq 0$ are calibration scalars. The result of Fig.2 reports that MixAMP is much faster than TFOCS in CPU runtime while providing comparable reconstruction quality in PSNR.

B. Separation of Direct and Finite-Difference Sparsity

We present another case by introducing finite-difference (FD) sparsity, then addressing a separation problem with a direct-and-FD sparse mixture. For the FD sparsity pursuit, we apply the *total variation* (TV) denoiser [10] which has been investigated by the numerous literature (see for example [12]). The TV denoiser is neither scalable nor block-separable because the FD sparsity cannot be defined by a single scalar of the signal, but depending upon all the adjacent of the scalar. Let $\mathbf{X}_b \in \mathbb{R}^{\sqrt{N}\times\sqrt{N}}$ have the FD sparsity. Then, we consider the TV denoiser, $\eta_b(\cdot) : \mathbb{R}^{\sqrt{N}\times\sqrt{N}} \to \mathbb{R}^{\sqrt{N}\times\sqrt{N}}$, which solves

$$\widehat{\mathbf{x}}_b = \eta_b(\mathbf{x}; \lambda) := \arg\min_{\mathbf{X}_b} ||\mathbf{X}_b||_{\text{TV}} + \frac{\lambda}{2}||\mathbf{X}_b - \mathbf{x}||_F^2. \qquad (14)$$

It is recognizable from (8) that the TV norm $||\mathbf{X}_b||_{\text{TV}}$ is the exponent of the sparsifying prior such that $\log f_{\mathbf{X}_b}(\mathbf{x}_b) = ||\mathbf{X}_b||_{\text{TV}}$. For the implementation of (14), external numerical solvers have been mostly considered since the TV norm is analytically non-scalable and non-smooth [10].

In Fig.3, we simultaneously estimate a shot noise $\mathbf{X}_a$ (10% sparsity) and a 128×128 *Cameraman* image $\mathbf{X}_b$ from the 2D measurements $\mathbf{Y}$ with 50% sampling ($M/N = 0.5$). In this example, we postulate that the *Cameraman* image has the FD sparsity and the shot noise is directly sparse. For the MixAMP separation, we apply the soft-thresholding (11) for $\mathbf{X}_a$, implementing the anisotropic TV denoising (14) for $\mathbf{X}_b$ using the 2D-Bregman iteration (Section 4 of [8]). In

50% Sampling for Y 10% sparsity of Xa $tol = 5\times10^{-4}$		
Solver Type	PSNR of Xb <dB>	Runtime <sec>
MixAMP	23.16	59.74
TFOCS	23.68	122.50

Fig. 3. An exemplary comparison of MixAMP and TFOCS in the direct-and-FD sparsity separation where we simultaneously estimate a shot noise $\mathbf{X}_a$ (10% sparsity) and a 128×128 *Cameraman* image $\mathbf{X}_b$ from the 2D measurements $\mathbf{Y}$ with 50% sampling ($M/N = 0.5$). Here, we set $\varepsilon = 10^{-10}, \lambda_1 = 2.0, \lambda_2 = 1.4$ for the TFOCS method (15).

the TFOCS method, we recast (3) to *Combining-L_1-and-TV minimization*, solving

$$\min \quad \lambda_1||\mathbf{X}_a||_1 + \lambda_2||\mathbf{X}_b||_{\text{TV}}$$
$$\text{s.t.} \quad ||\mathbf{Y} - \mathcal{P}_\Omega\{\mathbf{A}(\mathbf{X}_a + \mathbf{X}_b)\mathbf{A}^T\}||_F \leq \varepsilon. \qquad (15)$$

Likewise to the example in Section III-A, the result of Fig.3 validates the computational advantages of MixAMP where MixAMP is approximately 2 time faster than TFOCS for the same task. The scale difference of the runtime gap from the result of Section III-A is coming from the fact that the Bregman-TV denoiser (14) requires more computations than the block soft-thresholding (12) does.

These two comparison results support our claim that Mix-AMP outperforms the TFOCS method in computational efficiency under the 2D sparse signal separation task.

IV. CONCLUSIONS

In this letter, we have discussed the MixAMP method for the ill-posed L_1-*Split-Analysis*, applying the 2D sparse signal

separation problem. We first have developed the MixAMP method based on the factor graphical model of the 2D CS measurement model (1). Then, we have provided two cases of the study for the sparsity separation, validating the computational advantages of MixAMP through exemplary comparisons to the conventional first-order method, TFOCS [7]. Therefore, we claim that MixAMP is a very good alternative of the TFOCS method for solving the L_1-*Split-Analysis* in the 2D sparse separation problem.

SUPPLEMENTARY MATERIAL

A. 2D Compressed Sensing Model (1): A Practical Viewpoint

In this letter, we consider "the compressed sensing (CS) technique" with the model (1). The model (1) requires two-steps for the CS measurement generation of $\mathbf{Y}$ as shown in Fig.4: 1) linear mixing in two perpendicular axes by $\mathbf{AXA^T}$ and 2) undersampling by $\mathcal{P}_\Omega\{\cdot\}$. One can argue that this two-steps generation makes us to lose an advantage of CS which generates $\mathbf{Y}$ in the first place. Nevertheless, we state that this two-step generation is useful because of two practical reasons:

1) *For fast measurement generation with unitary matrices* $\mathbf{A}$: The model (1) can provide an accelerated generation of $\mathbf{Y}$ when the matrix $\mathbf{A}$ is some unitary types. For instance, this acceleration utilizes *Fast Cosine Transform* (FCT) when $\mathbf{A}$ is the DCT matrix, and *Fast Fourier Transform* when $\mathbf{A}$ is the Fourier matrix. Here, we consider the DCT case as an example. Let $\mathbf{A}_{\mathrm{DCT}} \in \mathbb{R}^{\sqrt{N} \times \sqrt{N}}$ denote the DCT matrix and $\mathbf{A}_{\mathrm{subDCT}} \in \mathbb{R}^{\sqrt{M} \times \sqrt{M}}$ is a sub-DCT matrix whose rows are randomly sampled from $\mathbf{A}_{\mathrm{DCT}}$. The conventional CS generation, expressed as

$$\mathbf{Y} = \mathbf{A}_{\mathrm{subDCT}} \, \mathbf{X} \, \mathbf{A}_{\mathrm{subDCT}}^T \in \mathbb{R}^{\sqrt{M} \times \sqrt{M}}, \quad (16)$$

spends $\mathcal{O}(NM^{\frac{1}{2}})$ computations due to the matrix multiplications. Compared to (16), the FCT-based generation with the model (1), *i.e.*,

$$\begin{aligned} \mathbf{Y} &= \mathcal{P}_\Omega\{\mathbf{A}_{\mathrm{DCT}} \, \mathbf{X} \, \mathbf{A}_{\mathrm{DCT}}^T\}, \\ &= \mathcal{P}_\Omega\{\mathrm{FCT}_{2D}[\mathbf{X}]\} \in \mathbb{R}^{\sqrt{N} \times \sqrt{N}}, \quad (17) \end{aligned}$$

is computationally efficient with $\mathcal{O}(N \log N^{\frac{1}{2}})$; therefore, its efficiency gets remarkable as the system size N increases. In addition, it is obvious that the FCT method is not applicable with the sub-DCT matrix $\mathbf{A}_{\mathrm{subDCT}}$.

2) *Use of a random decimation operator instead of* $\mathcal{P}_\Omega\{\cdot\}$: In practice, the random undersampling operator $\mathcal{P}_\Omega\{\cdot\}$ can be simply replaced by a random decimation operator. Then, the samples of $\mathbf{Y}$ can be stored to a memory with the size M by holding the knowledge of the set Ω. In this case, however, we need an inverse operator of the decimation operator to calculate $\mathbf{A}^T \mathbf{R} \mathbf{A}$ for the MixAMP method.

B. Hardship of Sparse Separation Problem

The sparsity separation, handled in this letter, is basically harder problem than the conventional single sparsity recovery.

This is because the sparsity separation includes not only the reconstruction of the mixture signal $\mathbf{X}$ from the CS measurements $\mathbf{Y}$, but also the separation of the two sparsity, $\mathbf{X}_a$ and $\mathbf{X}_b$, from the mixture $\mathbf{X}$. We can look at this hardship of the sparsity separation by simple manipulation from (1):

$$\mathbf{Y} = \mathcal{P}_\Omega \left\{ [\mathbf{A}|\mathbf{A}] \begin{bmatrix} \mathbf{X}_a \\ \mathbf{X}_b \end{bmatrix} \mathbf{A}^T \right\}, \quad (18)$$

From (18), we can reasonably conjecture that the reconstruction of the concatenated signal $[\mathbf{X}_a|\mathbf{X}_b]^T$ from $\mathbf{Y}$ requires larger M than the single sparsity recovery does. In addition, while revising this letter, we became aware of a theoretical work of Studer *et al.* [16] which supports our conjecture by providing a coherence-based sufficient condition for recovery guarantees of $\mathbf{X}_a$ and $\mathbf{X}_b$.

Naturally, the reconstruction quality of the sparse separation gets better as the sampling rate M/N increases. In Fig.5, we provide an extended result of the $\mathbf{X}_b$ reconstruction in the directed-and-FD sparsity separation (Section III-B) for a variety of M/N. From the figure, we see that the both methods improve their reconstruction quality given a higher M/N. This result also shows that although the TFOCS method has a small lead in the reconstruction quality as M/N becomes higher, the MixAMP method is still remarkably advantageous in computational cost.

Fig. 4. An example of 2D CS measurement generation in the model (1), where we consider the DCT matrix $\mathbf{A}$ and the 128×128 *Cameraman* image as the 2D target signal $\mathbf{X}$. First, the signal $\mathbf{X}$ is linearly mixed in two perpendicular axes by the DCT matrix $\mathbf{A}$, generating $\mathbf{A}\mathbf{X}\mathbf{A}^{\mathbf{T}}$. Second, the measurement $\mathbf{Y}$ is produced by applying the undersampling operator $\mathcal{P}_{\Omega}\{\cdot\}$ to $\mathbf{A}\mathbf{X}\mathbf{A}^{\mathbf{T}}$.

Fig. 5. Extended numerical result of the directed-and-FD sparsity separation in Section III-B: In this figure, we only plot reconstruction of the 128×128 *Cameraman image* $\mathbf{X}_b$ for a variety of the sampling rate M/N. The others for the experimental setup remains the same as Section III-B. For the TFOCS method (15), we use empirically-tuned parameter set $(\lambda_1, \lambda_2, \varepsilon)$.

REFERENCES

[1] N. Zhou, H. Li, D. Wang, S. Pan, and Z. Zhou, "Image compression and encryption scheme based on 2D compressive sensing and fractional Mellin transform," *Optics Communi.*, vol. 343, pp. 10-21, May. 2015.

[2] M. F. Duarte, and R. G. Baraniuk, "Kronecker compressive sensing," *IEEE Trans. Image. Process.* vol. 21, no. 2, pp. 494-504, Feb. 2012

[3] E. J. Candes, Y. C. Eldar, D. Needell, and P. Randall, "Compressed sensing with coherent and redundant dictionaries," *Appl. Comput. Harmon. Anal.*, vol. 31, no. 1, pp. 59-73, Sep. 2011.

[4] C. Aubel, C. Studer, G. Pope, H. Bolcskei, "Sparse signal separation in redundant dictionaries," in *Proc. IEEE Intl. Symp. on Inf. Theory, ISIT*, Boston, MA, USA, pp. 2047-2051, 2012.

[5] J.-F. Cai, S. Osher, and Z. Shen, "Split Bregman methods and frame based

image restoration," *Multiscale Modeling and Simulation*, vol. 8, no. 2, pp. 337-369, Jan. 2010.

[6] S. G. Mallat and G. Yu, "Super-resolution with sparse mixing estimators," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2889-2900, Nov. 2010.

[7] S. Becker, E. J. Candes and M. Gran, "Templates for Convex Cone Problems with Applications to Sparse Signal Recovery," *Math. Program. Comput.*, vol. 3, num. 3, Aug. 2011. MATLAB code available at http://cvxr.com

[8] T. Goldstein and S. Osher. "The split Bregman algorithm for l1 regularized problems," *SIAM J. Imaging Sciences*, vol. 2, pp. 323-343, Apr. 2009.

[9] D.L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing," *Proc. Nat. Acad. Sci.*, vol. 106, pp. 18914-18919, Nov. 2009.

[10] D.L. Donoho, I. Johnstone, and A. Montanari, "Accurate prediction of phase transitions in compressed sensing via a connection to minimax denoising, " *IEEE Trans. Inform. Theory*, vol. 59, no. 6, pp. 3396-3433, June 2013.

[11] A. Montanari, "Graphical models concepts in compressed sensing," available at arXiv:1011.4328v3[cs.IT], Mar. 2011.

[12] L.I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D:Nonlinear Phenomena*, vol. 60, pp. 259-268, Nov. 1992.

[13] J. Starck, M. Elad, and D. Donoho,"Image decomposition via the combination of sparse representations and a variational approach,"

[14] J. Bobin, J. L. Starck, J. M. Fadili, Y. Moudden, and D. L. Donoho, "Morphological component analysis: An adaptive thresholding strategy," *IEEE Trans. Image Process.*, vol. 16, no. 11, pp. 2675-2681, Nov. 2007.

[15] J. Ma, G. Plonka, M. Y. Hussaini, "Compressive video sampling with approximate message passing decoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 9, pp. 1354-1364, Sep. 2012.

[16] C. Studer, P. Kuppinger, G. Pope, and H. Bolcskei, "Recovery of sparsely corrupted signals," *IEEE Trans. Inform. Theory*, vol. 58, no. 5, pp. 3115-3130, May 2012.