Effective Snapshot Compressive-spectral Imaging via Deep Denoising and Total Variation Priors

Haiquan Qiu[1], Yao Wang[1,2]*, Deyu Meng[1,3]
[1]Xi'an Jiaotong University, Xi'an, China
[2]Shanghai Em-Data Technology Co., Ltd., China
[3]The Macau University of Science and Technology, Macau, China
qiuhaiquan@stu.xjtu.edu.cn, yao.s.wang@gmail.com, dymeng@mail.xjtu.edu.cn

Abstract

Snapshot compressive imaging (SCI) is a new type of compressive imaging system that compresses multiple frames of images into a single snapshot measurement, which enjoys low cost, low bandwidth, and high-speed sensing rate. By applying the existing SCI methods to deal with hyperspectral images, however, could not fully exploit the underlying structures, and thereby demonstrate unsatisfactory reconstruction performance. To remedy such issue, this paper aims to propose a new effective method by taking advantage of two intrinsic priors of the hyperspectral images, namely deep image denoising and total variation (TV) priors. Specifically, we propose an optimization objective to utilize these two priors. By solving this optimization objective, our method is equivalent to incorporate a weighted FFDNet and a 2DTV or 3DTV denoiser into the plug-and-play framework. Extensive numerical experiments demonstrate the outperformance of the proposed method over several state-of-the-art alternatives. Additionally, we provide a detailed convergence analysis of the resulting plug-and-play algorithm under relatively weak conditions such as without using diminishing step sizes. The code is available at `https://github.com/ucker/SCI-TV-FFDNet`.

1. Introduction

Compressive sensing [5, 1] is a popular imaging technology that can be employed to capture video [8, 19, 15, 32, 22, 23] and hyperspectral images [6, 24, 25, 30, 2]. One of the most important compressive sensing systems is the so-called snapshot compressive imaging (SCI)[15, 24]. Precisely, SCI uses 2D sensors to obtain higher dimensional image data and exploit corresponding algorithms to reconstruct the desired data. As compared with traditional compressive sensing technology, SCI possesses of low memory, low power consumption, low bandwidth and low cost, and as such, can be used to efficient capture the hyperspectral images. Among the existing SCI systems, coded aperture snapshot spectral imaging (CASSI)[25] is a representative hyperspectral SCI system, which combines hyperspectral images of different wavelengths into a single 2D one.

Along with the development of hardware, various reconstruction algorithms have been proposed for SCI. GAP-TV [28] applied total variation minimization under the generalized alternating projection (GAP) [13] framework. Recently, DeSCI [14] demonstrates the-state-of-art results in reconstructing both video and hyperspectral image data. As further shown in [31], DeSCI can be regarded as a plug-and-play (PnP) algorithm that employs rank minimization as an intermediate step during reconstruction. However, the low computational speed of DeSCI precludes its applications. For example, DeSCI costs more than six hours to reconstruct a hyperspectral image of size $1021 \times 703 \times 24$ from its snapshot measurement. While GAP-TV is a faster algorithm, it cannot reconstruct high-quality images that can be fitted for real applications. Therefore, [31] incorporated a deep denoiser network such as FFDNet [34] into PnP algorithm [3]. Because FFDNet can be performed on GPU, it runs very fast compared with DeSCI. However, [31] mainly focused on video SCI reconstruction, and as we shall show later, its reconstruction performance on hyperspectral images are not satisfactory.

Basically, applying DeSCI for hyperspectral image reconstruction requires to perform GAP-TV to get its initial value. Numerical experiments revealed that this initial value is crucial for the performance of DeSCI. If the initialization is slightly worse, the performance of DeSCI could be largely poor. As such, it is highly demanding to develop newly effective method to address the aforementioned issues.

We regard this initialization as a primitive method us-

*Corresponding author.

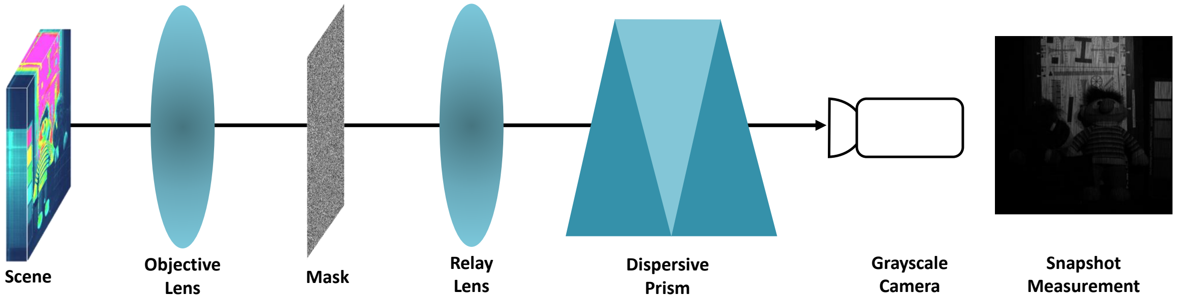

Figure 1. Sensing process of CASSI.

ing different priors. Considering that different intrinsic priors could promote each other, we propose a new method by combining the deep image denoising and TV priors to enhance hyperspectral image reconstruction. Although using a denoising network such as FFDNet or TV regularization alone can not obtain satisfactory results, the priors combined by our method can take advantage of denoising network and TV at the same time and make the two priors promote each other, and thereby obtain more much better reconstruction results.

Our paper makes the following contributions:

1. We propose a general framework that can combine deep denoising and total variation priors for SCI reconstruction of hyperspectral images. In the reconstruction process, these two priors can promote each other to boost the quality of reconstructed images.

2. We conduct extensive experiments on both simulated and real datasets. The numerical results have verified the efficiency and effectiveness of our algorithm.

3. We prove the convergence of our algorithm for SCI reconstruction without using diminishing step sizes. What's more, we are the first ones that prove the convergence of accelerated PnP-GAP for SCI, which is used in our simulated experiments.

The rest of this paper is organized as follows. We introduce the SCI model and the corresponding reconstruction algorithms in Section 2. Our proposed method and the convergence analysis of the resulting algorithm are developed in Section 3. Extensive experiments are presented in Section 4.

Related work Different priors need to be employed if we want to reconstruct images from SCI system. A variety of algorithms have been proposed for SCI reconstruction, such as sparsity-based algorithm [29, 33], GMM [26, 30], GAP-TV [28] and DeSCI [14]. Among them, DeSCI has led to state-of-the-art results. In addition to explicit modeling of the priors of SCI problem, implicit priors are also employed for reconstruction. [31] integrates various denoisers such as FFDNet [34] into PnP algorithm for video SCI reconstruction and obtain excellent reconsctruction results. Most recently, some deep learning methods also achieves good results for SCI problem [11, 12, 17, 18, 21, 16]. Different from these methods, we combine deep denoising and total variation priors into PnP framework to boost SCI reconstruction of hyperspectral images.

[31] proves the convergence of PnP-GAP for SCI reconstruction using diminishing step sizes. Recently, [20] proves the convergence of PnP-ADMM without using diminishing step sizes. Inspired by these two works, we prove the fixed point convergence of PnP-GAP and accelerated PnP-GAP without using diminishing step sizes.

2. Review of Snapshot Compressive-spectral Imaging

2.1. SCI model

CASSI is a representative SCI system for capturing hyperspectral image. Figure 1 shows the sensing process of CASSI. The fixed mask spatially encodes the spectral scene. The encoded scene is spectrally dispersed by the prism. Finally, a grayscale camera detects the spatial spectrum encoding scene. Therefore, the snapshot on the camera could encode dozens of spectral bands of the scene.

So we consider that a B sensing masks $\mathbf{C} \in \mathbb{R}^{n_x \times n_y \times B}$ compresses and modulates a B-bands spectral image $\mathbf{X} \in \mathbb{R}^{n_x \times n_y \times B}$ into the measurements $\mathbf{Y} \in \mathbb{R}^{n_x \times n_y}$. This process can be mathematically expressed as

$$\mathbf{Y} = \sum_{b=1}^{B} \mathbf{C}_b \odot \mathbf{X}_b + \mathbf{Z}, \tag{1}$$

where $\mathbf{C}_b = \mathbf{C}(:,:,b)$ and $\mathbf{X}_b = \mathbf{X}(:,:,b) \in \mathbb{R}^{n_x \times n_y}$ represent the b-th sensing mask and the image of the corresponding hyperspectral band, respectively; $\mathbf{Z} \in \mathbb{R}^{n_x \times n_y}$ denotes the noise term; $\odot$ denotes the Hadamard (element-wise) product.

We can rewrite Eq. (1) in matrix-vector product form

$$as\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{z}, \tag{2}$$

where $\mathbf{y} = \text{Vec}(\mathbf{Y}) \in \mathbb{R}^{n_x n_y}$ and $\mathbf{z} = \text{Vec}(\mathbf{Z}) \in \mathbb{R}^{n_x n_y}$. The hyperspectral image vector $\mathbf{x} \in \mathbb{R}^{n_x n_y B}$ is expressed as

$$\mathbf{x} = \text{Vec}(\mathbf{X}) = [\text{Vec}(\mathbf{X}_1)^T, ..., \text{Vec}(\mathbf{X}_B)^T]^T. \tag{3}$$

Unlike traditional compressive sensing [5], the sensing matrix $\mathbf{H} \in \mathbb{R}^{n_x n_y \times n_x n_y B}$ in hyperspectral image SCI is not a dense matrix. The special structure of $\mathbf{H}$ can be denoted as

$$\mathbf{H} = [\mathbf{D}_1, ..., \mathbf{D}_B]. \tag{4}$$

where $\mathbf{D}_b = \text{diag}(\text{Vec}(\mathbf{C}_b)) \in \mathbb{R}^{n \times n}$ with $n = n_x n_y$, for $b = 1, \ldots B$. Because $\mathbf{x} \in \mathbb{R}^{n_x n_y B}$ and $\mathbf{H} \in \mathbb{R}^{n_x n_y \times n_x n_y B}$, the sampling rate of SCI is equal to $1/B$. [9, 10] have theoretically proved that the recovery of $\mathbf{x}$ from $\mathbf{y}$ is possible when $B > 1$.

2.2. Plug-and-Play Algorithms for SCI reconstruction

The mathematical model of SCI can be expressed as the following inverse problem:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2 + \lambda g(\mathbf{x}), \tag{5}$$

where $g(\mathbf{x})$ is the regularization, λ is the tuning parameter of regularization. We shall introduce two algorithms for image reconstruction, i.e., PnP-GAP and PnP-ADMM [31]. They are equivalent in the noiseless conditions[14]. In this section, we simply give the concrete step of these algorithms. If one want to know how to derive these steps, please refer to [14].

PnP-ADMM for SCI PnP-ADMM has the following form:

$$\mathbf{x}^{(k+1)} = (\mathbf{H}^T\mathbf{H} + \gamma\mathbf{I})^{-1}[\mathbf{H}^T\mathbf{y} + \gamma(\mathbf{v}^{(k)} + \mathbf{u}^{(k)})], \tag{6}$$

$$\mathbf{v}^{(k+1)} = \mathcal{D}_\sigma(\mathbf{x}^{(k+1)} - \mathbf{u}^{(k)}) \tag{7}$$

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + (\mathbf{v}^{(k+1)} - \mathbf{x}^{(k+1)}), \tag{8}$$

where the superscript (k) denotes the iteration number.

PnP-GAP for SCI PnP-GAP has the following form:

$$\mathbf{x}^{(k+1)} = \mathbf{v}^{(k)} + \mathbf{H}^T(\mathbf{H}\mathbf{H}^T)^{-1}(\mathbf{y} - \mathbf{H}\mathbf{v}^{(k)}), \tag{9}$$

$$\mathbf{v}^{(k+1)} = \mathcal{D}_\sigma(\mathbf{x}^{(k+1)}). \tag{10}$$

PnP-GAP first projects data on linear manifold $\mathbf{y} = \mathbf{H}\mathbf{x}$, then denoises the projected data. In the noiseless condition, accelerated PnP-GAP is proposed as follows:

$$\begin{aligned}
\mathbf{y}^{(k+1)} &= \mathbf{y}^{(k)} + \mathbf{y} - \mathbf{H}\mathbf{v}^{(k+1)}, \\
\tilde{\mathbf{x}}^{(k+2)} &= \mathbf{v}^{(k+1)} + \mathbf{H}(\mathbf{H}\mathbf{H}^T)^{-1}\left(\mathbf{y}^{(k+1)} - \mathbf{H}\mathbf{v}^{(k+1)}\right), \\
\mathbf{v}^{(k+2)} &= \mathcal{D}_\sigma(\tilde{\mathbf{x}}^{(k+2)}).
\end{aligned}$$

And accelerated PnP-GAP performs much better than PnP-GAP in the noiseless condition. The relationship between PnP-GAP and accelerated PnP-GAP is discussed in [14]. [31] further introduced the relationship between PnP-ADMM and PnP-GAP.

3. Combining Deep Denoising and Total Variation Priors

As mentioned before, DeSCI for hyperspectral image reconstruction employs the GAP-TV result as its initial value, which is critical to its performance. However, this way of combining different priors is relatively primitive. We will introduce our way of utilizing different priors in this section. We shall first introduce the basic idea behind our method. Assume that there exists a best prior in each step of SCI reconstruction algorithm. If the optimization algorithms with different priors have similar reconstruction processes[1], their best priors for each iteration should be similar in the reconstruction processes. Assume that there have two priors we want to combine, our method chooses the prior closest to these two priors, then such a prior is likely to be close to the best prior. Intuitively, the closeness of different priors to each other gives them a chance to possess the advantages of each other. Motivated by this, we propose a general framework based on PnP algorithm to combine two different priors namely FFDNet and TV. The basic idea of this method is shown in Figure 2.

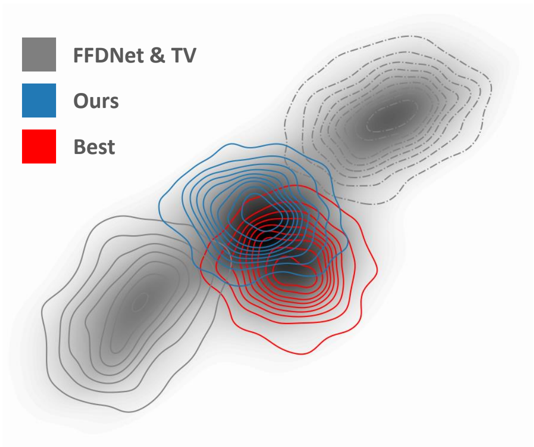

Figure 2. Our method would like to combine the two posteriors $p(\mathbf{v}|\mathbf{x})$ and $q(\mathbf{v}|\mathbf{x})$ corresponding to FFDNet (gray line) and TV (gray dash-dot line). These posteriors are determined by the distribution of hyperparameter $p(\sigma|\mathbf{x})$ and $q(t|\mathbf{x})$. FFDNet and TV with random distributions $p(\sigma|\mathbf{x})$ and $q(t|\mathbf{x})$ are away from the best posterior (red line). Our method (blue line) finds the closest posterior between FFDNet and TV posterior which is closer to the best posterior.

[1] The reconstructed images are similar in each iteration.

In the following Section 3.1, we shall treat the denoising step in the PnP algorithm as MAP estimation, derive the posterior[2] for denoiser, and put forward the optimization objective of our method.

3.1. Denoiser into Posterior

The denoising step $\mathbf{v}^{(k+1)} = \mathcal{D}_\sigma(\mathbf{x}^{(k+1)})$ is equivalent to solve the following problem:

$$\mathbf{v}^{(k+1)} = \arg\min_{\mathbf{v}} \frac{\lambda}{\gamma} g(\mathbf{v}) + \frac{1}{2}\|\mathbf{x}^{(k+1)} - \mathbf{v}\|_2^2, \tag{11}$$

where $g(\mathbf{v})$ is the regularization term. The proximal operator of $g(\mathbf{v})$ can be employed as FFDNet and TV denoiser. From the probability perspective, Eq. (11) can be regarded as maximum a posterior (MAP) estimation

$$\begin{aligned} \mathbf{v}^{(k+1)} &= \arg\max_{\mathbf{v}} p(\mathbf{v}|\mathbf{x}^{(k+1)}, \sigma) & (12) \\ &= \mathcal{D}_\sigma(\mathbf{x}^{(k+1)}). & (13) \end{aligned}$$

Now we assume there exists a distribution about the denoiser hyperparameter σ and denote it as $p(\sigma|\mathbf{x}^{(k+1)})$. With the distribution of hyperparameter, we integrate σ to eliminate the hyperparameter as

$$p(\mathbf{v}|\mathbf{x}^{(k+1)}) = \int p(\mathbf{v}|\mathbf{x}^{(k+1)}, \sigma) p(\sigma|\mathbf{x}^{(k+1)}) d\sigma. \tag{14}$$

It is easy to see that Eq. (14) is the posterior we get from the denoiser $\mathcal{D}_\sigma$.

Since our method is proposed to combine two denoisers, we denote another denoiser as $\mathcal{D}_t$. Doing the same to $\mathcal{D}_t$ as above, we have

$$q(\mathbf{v}|\mathbf{x}^{(k+1)}) = \int q(\mathbf{v}|\mathbf{x}^{(k+1)}, t) q(t|\mathbf{x}^{(k+1)}) dt. \tag{15}$$

Then our goal is to minimize the distance between $p(\mathbf{v}|\mathbf{x}^{(k+1)})$ and $q(\mathbf{v}|\mathbf{x}^{(k+1)})$, that is,

$$\min_{p(\sigma|\mathbf{x}^{(k+1)}), q(t|\mathbf{x}^{(k+1)})} \operatorname{dist}\left(p(\mathbf{v}|\mathbf{x}^{(k+1)}), q(\mathbf{v}|\mathbf{x}^{(k+1)})\right), \tag{16}$$

where $\operatorname{dist}(\cdot,\cdot)$ is the distance function. To minimize such distance, we list two challenges and their corresponding solutions:

- $p(\mathbf{v}|\mathbf{x}^{(k+1)}, \sigma)$ and $q(\mathbf{v}|\mathbf{x}^{(k+1)}, t)$ are unknown distributions, so we have to model them;
- $p(\mathbf{v}|\mathbf{x}^{(k+1)})$ and $q(\mathbf{v}|\mathbf{x}^{(k+1)})$ are difficult to calculate and we will discretize the integral in Eq. (14) and Eq. (15).

Next we will resolve these challenges and make problem (16) solvable.

3.2. Minimizing Distance between Posteriors

We will resolve the two challenges mentioned above to minimize the distance between posteriors in this section. The denoisers corresponding to the two posteriors $p(\mathbf{v}|\mathbf{x})$ and $q(\mathbf{v}|\mathbf{x})$[3] are FFDNet and TV. First we will model the distribution $p(\mathbf{v}|\mathbf{x}, \sigma)$ and $q(\mathbf{v}|\mathbf{x}, t)$.

It is reasonable to model the posterior $p(\mathbf{v}|\mathbf{x}, \sigma)$ of FFDNet as Gaussian distribution, since the training data of FFDNet is a pair of clean and noisy image and the noise distribution is Gaussian. We model the posterior corresponding to FFDNet as

$$p(\mathbf{v}|\mathbf{x}, \sigma) = N(\mathrm{FFD}_\sigma(\mathbf{x}), \sigma\mathbf{I}), \tag{17}$$

where $\mathbf{I}$ is the identity matrix, $\mathrm{FFD}_\sigma(\mathbf{x})$ is FFDNet which takes $\mathbf{x}$ as its input. In Eq. (17), $\mathrm{FFD}_\sigma(\mathbf{x})$ is the mean, $\sigma\mathbf{I}$ is the covariance matrix of the Gaussian distribution. $\mathrm{FFD}_\sigma(\mathbf{x})$ is in $\mathbb{R}^{n_x \times n_y \times B}$ if we aggregate the denoising results of all spectral bands.

We also use Gaussian distribution to model the posterior for TV denoiser for convenient. And we have

$$q(\mathbf{v}|\mathbf{x}, t) = N(\mathrm{TV}_t(\mathbf{x}), \Sigma_t), \tag{18}$$

where $\mathrm{TV}_t(\mathbf{x}) \in \mathbb{R}^{n_x \times n_y \times B}$ is the mean, Σ_t is the covariance matrix. The mean of $q(\mathbf{v}|\mathbf{x}, t)$ is the TV denoising result of $\mathbf{x}$. For simplicity, we assume Σ_t is a constant matrix. As you will see in the latter part of this section, the actual value of Σ_t doesn't matter in our algorithm, so we don't spend more effort on modeling the specific variance of $q(\mathbf{v}|\mathbf{x}, t)$.

Next we are going to resolve the second challenge. In essence, the difficulty of calculating $p(\mathbf{v}|\mathbf{x})$ and $q(\mathbf{v}|\mathbf{x})$ comes from the continuity of σ and t. Therefore, a very straightforward method is to discretize σ and t. To discretize σ, we have $\sigma \in A$ where A is a set with finite elements and $p(\sigma|\mathbf{x})$ is a discrete distribution. Doing the same to t, we have $t \in B$ where B is also a set with finite elements. Now we have two discrete distributions $p(\sigma|\mathbf{x})$ and $q(t|\mathbf{x})$. Then Eq. (14) and Eq. (15) can be rewritten as

$$\begin{aligned} p(\mathbf{v}|\mathbf{x}) &= \sum_{\sigma \in A} p(\mathbf{v}|\mathbf{x}, \sigma) p(\sigma|\mathbf{x}), \\ q(\mathbf{v}|\mathbf{x}) &= \sum_{t \in B} p(\mathbf{v}|\mathbf{x}, t) p(t|\mathbf{x}). \end{aligned}$$

Now we need to specify the distance function $\operatorname{dist}(\cdot,\cdot)$ in problem (16). We have tried various metrics as the distance in experiments, such as KL divergence, L_2 distance, and MMD[7]. But in this paper, we choose to use the L_2 distance between the first moments of $p(\mathbf{v}|\mathbf{x})$ and $q(\mathbf{v}|\mathbf{x})$ because of computational efficiency and good experimental

[2] We do some clarification on words *posterior* and *prior* here. Posterior has a one-to-one correspondence to denoiser. And denoiser is an implicit prior in our method. Therefore, posterior and prior is the same thing to some extend here.

[3] We omit the superscript $(k+1)$ for simplicity.

results. In other words, we want the mean of $p(\mathbf{v}|\mathbf{x})$ and $q(\mathbf{v}|\mathbf{x})$ to be close. Therefore, we come to the following optimization problem

$$\min_{p(\sigma|\mathbf{x}),q(t|\mathbf{x})} \|\mathbb{E}_{p(\mathbf{v}|\mathbf{x})}[\mathbf{v}] - \mathbb{E}_{q(\mathbf{v}|\mathbf{x})}[\mathbf{v}]\|_2^2$$

We rearrange the above optimization problem and denote $w_\sigma^{ffd} = p(\sigma|\mathbf{x}), \sigma \in A$ and $w_t^{tv} = q(t|\mathbf{x}), t \in B$ to get the following optimization problem

$$\begin{aligned} \min_{w^{ffd},w^t} \ & \| \sum_{\sigma \in A} w_\sigma^{ffd} \mathrm{FFD}_\sigma(\mathbf{x}) - \sum_{t \in B} w_t^{tv} \mathrm{TV}_t(\mathbf{x}) \|_2^2 \quad (19) \\ \text{subject to } & \sum_{\sigma \in A} w_\sigma^{ffd} = 1, \sum_{t \in B} w_t^{tv} = 1, \\ & w_\sigma^{ffd} \geq 0, w_t^{tv} \geq 0, \sigma \in A, t \in B. \end{aligned}$$

where $w^{ffd} = \{w_\sigma^{ffd} : \sigma \in A\}$ and $w^{tv} = \{w_t^{tv} : t \in B\}$. The optimization problem (19) can be rewritten as

$$\begin{aligned} & \min_W W^T \mathbf{P} W \quad (20) \\ \text{subject to } & \sum_{i=|A|+1}^{|A|+|B|} W_i = 1 \\ & \sum_{i=1}^{|A|} W_i = 1, W \geq 0. \end{aligned}$$

where $W = [w_{\sigma_1}^{ffd}, \cdots, w_{\sigma_{|A|}}^{ffd}, w_{t_1}^{tv}, \cdots, w_{t_{|B|}}^{tv}]^T$. We further let $X_f = [\mathrm{FFD}_{\sigma_1}(\mathbf{x}), \cdots, \mathrm{FFD}_{\sigma_{|A|}}(\mathbf{x})]$, $X_t = [\mathrm{TV}_{t_1}(\mathbf{x}), \cdots, \mathrm{TV}_{t_{|B|}}(\mathbf{x})]$, then the semi-positive definite matrix $\mathbf{P}$ is denoted as

$$\mathbf{P} = \begin{bmatrix} X_f^T X_f & -X_f^T X_t \\ -X_t^T X_f & X_t^T X_t \end{bmatrix} \quad (21)$$

The optimization problem (20) is quadratic programming. Since $|A| + |B|$ is relatively small in our experiment, the problem (20) can be quickly solved.

By solving problem (20), we have two similar posteriors that may have the advantages of each other. Then we simply combine these posteriors on average and get a weighted denoiser from the mean of the combined posterior[4]. And this weighted denoiser is employed in plug-and-play algorithms. We combine two posteriors $p(\mathbf{v}|\mathbf{x})$ and $q(\mathbf{v}|\mathbf{x})$ on average for fairness because we can not tell which one is better than the other without extra information.

3.3. Algorithm

We can apply our method to any plug-and-play algorithm. We take PnP-GAP as an example in this section. Our algorithm has four steps:

1. Euclidean projection. This procedure is the same as Eq.(9);
2. Obtain denoising image based on the hyperparameters in the sets A and B. Denote them as $\{\mathbf{v}_\sigma : \sigma \in A\}$ and $\{\mathbf{v}_t : t \in B\}$;
3. Solve problem (20). Get the best weighting coefficient $\hat{w}^{ffd}$ and $\hat{w}^{tv}$;
4. Obtain the denoising result

$$\mathbf{v}^{(k+1)} = \frac{1}{2}(\sum_{\sigma \in A} \hat{w}_\sigma^{ffd} \mathbf{v}_\sigma + \sum_{t \in B} \hat{w}_t^{tv} \mathbf{v}_t). \quad (22)$$

Compared with the classic PnP-GAP, our algorithm needs to perform $|A| + |B|$ steps for denoising and solve a quadratic programming problem. The pseudo-code of our method is illustrated in Algorithm 1.

Algorithm 1 our proposed Plug-and-Play GAP

```
Require: H, y.
Initial $\mathbf{v}^{(0)}$, $A$, $B$
while Not Converge do
    Update $\mathbf{x}$ by Eq. (9).
    Obtain denoising image set $\{\mathbf{v}_\sigma : \sigma \in A\}$ by $\mathbf{v}_\sigma^{(k+1)} = \mathrm{FFD}_\sigma(\mathbf{x}^{(k+1)})$.
    Obtain denoising image set $\{\mathbf{v}_t : \sigma \in B\}$ by $\mathbf{v}_t^{(k+1)} = \mathrm{TV}_t(\mathbf{x}^{(k+1)})$.
    Solve optimization problem (20)
    Update $\mathbf{v}$ by Eq. (22)
end while
```

3.4. Fixed-point Convergence

Motivated by [20] and [31], we can prove our proposed PnP-GAP converges to a fixed point. Different from [31], we prove the convergence without using diminishing step sizes. What's more, we also prove the convergence of accelerated PnP-GAP which is employed in our simulated experiments. First, we make Assumption 1 about the denoiser. Because our proposed algorithm is equivalent to employ weighted denoiser in plug-and-play algorithm, we prove that the weighted denoiser also meets Assumption 1 in Lemma 1. Then we convert proposed PnP-GAP and accelerated PnP-GAP into two operators respectively. Theorem 1 and Theorem 2 state that their operators are contractions in some conditions.

We first introduce the assumption of the denoisers used in our paper.

[4] In fact, we should perform maximum a posterior estimation for the combined posterior. However, the combined posterior is Gaussian Mixture distribution and MAP for Gaussian mixture distribution is non-convex problem which is hard to optimize. We find that the mean of Gaussian mixture distribution is equivalent to weighted denoisers which is reasonable in common sense and has good experimental results.

Assumption 1 (Assumption (A) in [20])**.** *We assume that all denoisers $\mathcal{D}_\sigma : \mathbb{R}^d \mapsto \mathbb{R}^d$ used in our method satisfy*

$$\|(\mathcal{D}_\sigma - \mathbf{I})(x) - (\mathcal{D}_\sigma - \mathbf{I})(y)\|_2 \leq \epsilon \|x - y\|_2 \tag{23}$$

for all $x, y \in \mathbb{R}^d$ for some $\epsilon > 0$.

We can choose small σ such that $\mathcal{D}_\sigma$ is close to identity mapping. Therefore, Assumption 1 is reasonable. If all denoisers meet Assumption 1, it can be proved that the weighted denoiser also meets Assumption 1.

Lemma 1. *$\mathcal{S} = \{\mathcal{D}_\sigma : \sigma \in S\}$ is a set of denoiser satisfying Assumption 1 and $|\mathcal{S}| < \infty$. Then the weighted denoiser of $\mathcal{S}$:*

$$\mathcal{D}_w(x) = \sum_{\sigma \in S} w_\sigma \mathcal{D}_\sigma(x)$$

also satisfies Assumption 1, where $\sum_{\sigma \in S} w_\sigma = 1, w_\sigma \geq 0, \forall \sigma \in S$.

See the proof in the supplementary material.

In order to prove the convergence, we need the following assumption.

Assumption 2 (Assumption 1 in [31])**.** *Assume that $\{R_j\}_{j=1}^n > 0$ which means for each spatial location j, the B-frame modulation masks at this location have at least one non-zero entries. We further assume $R_{\max} > R_{\min}$*

In Assumption 2, $R = \mathbf{H}\mathbf{H}^T = \text{diag}(R_1, \cdots, R_n)$ and we define

$$\begin{aligned} R_{\max} &:= \max(R_1, \cdots, R_n) = \lambda_{\max}(\mathbf{H}\mathbf{H}^T) \\ R_{\min} &:= \min(R_1, \cdots, R_n) = \lambda_{\min}(\mathbf{H}\mathbf{H}^T) \end{aligned}$$

where $\lambda_{\max}(\mathbf{H}\mathbf{H}^T)$ and $\lambda_{\min}(\mathbf{H}\mathbf{H}^T)$ represent maximum and minimum eigenvalues of $\mathbf{H}\mathbf{H}^T$ respectively.

3.4.1 Convergence of PnP-GAP

We denote P as the Euclidean projection and $\mathcal{D}_\sigma$ as denoiser. Next theorem states the convergence of PnP-GAP.

Theorem 1. *Assume $\mathbf{H}$ satisfies Assumption 2. Then the following operator*

$$G = \mathcal{D}_\sigma \circ P$$

is a contraction if $\mathcal{D}_\sigma$ satisfies Assumption 1 and

$$0 < \epsilon < \sqrt{\frac{R_{\max}}{R_{\max} - R_{\min}}} - 1.$$

Remark 1. *In Theorem 1, G first projects data and then denoises the projected data which is equivalent to PnP-GAP. G being a contraction means PnP-GAP converges to a fixed point.*

See the proof of Theorem 1 in the supplementary material.

3.4.2 Convergence of accelerated PnP-GAP

To prove the convergence of accelerated PnP-GAP, we need to prove the following operator is a contraction.

$$T = \frac{1}{2}\mathbf{I} + \frac{1}{2}(2P - \mathbf{I})(2\mathcal{D}_\sigma - \mathbf{I}) \tag{24}$$

where P is the Euclidean projection onto the linear manifold $\mathbf{y} = \mathbf{Hx}$ and $\mathcal{D}_\sigma$ is a denoiser. $z^{(k+1)} = T(z^{(k)})$ is the PnP-DRS(plug-and-play Douglas–Rachford splitting) whose convergence is equivalent to the convergence of PnP-ADMM[5]. And the convergence of PnP-ADMM is equivalent to the convergence of accelerated PnP-GAP (these equivalences are presented in supplementary material). Now, the following theorem says that T is a contraction.

Theorem 2. *Assume $\mathbf{H}$ satisfies Assumption 2. Let P be a Euclidean projection on linear manifold $\mathbf{y} = \mathbf{Hx}$. Then*

$$T = \frac{1}{2}\mathbf{I} + \frac{1}{2}(2P - \mathbf{I})(2\mathcal{D}_\sigma - \mathbf{I})$$

is a contraction if $\mathcal{D}_\sigma$ satisfies Assumption 1 and

$$0 < \epsilon < 1 - \sqrt{1 - \frac{R_{\min}}{R_{\max}}}.$$

Remark 2. *From the supplementary material, we know that there are the following relationships between different algorithms:*

Convergence of PnP-DRS

$\Rightarrow$ *Convergence of PnP-ADMM*

$\Rightarrow$ *Convergence of accelerated PnP-GAP.*

Theorem 2 has proved the convergence of PnP-DRS, which means accelerated PnP-GAP converges to a fixed point.

See the proof of Theorem 2 in the supplementary material.

4. Experiments

In this section, we compare our proposed method with several methods such as GAP-TV and DeSCI. Because our algorithm combines TV and FFDNet, we also compare PnP-FFDNet-TV (denoted as FFDNet-TV in tables and FFD_TV in figures) which first performs 50 iteration FFDNet and then 50 iteration TV[6]. PnP-FFDNet-TV is a primitive way of combining different priors. The performance of different algorithms is evaluated by two indicators: peak signal-to-noise ratio (PSNR) and structural

[5] PnP-ADMM here is different as it updates $\mathbf{x}^{(k+1)}$ with Eq. (9) instead of Eq. (6).

[6] The results of first performing TV denoising are worse.

Table 1. The results of PSNR in dB (left entry in each cell) and SSIM (right entry in each cell) by different algorithms on Bird and Toy.

Data	2DTV	3DTV	FFDNet	DeSCI	FFDNet-TV	Ours (2DTV)	Ours (3DTV)
Toy	25.26, 0.8630	28.46, 0.9102	24.28, 0.8298	26.62, 0.9116	25.49, 0.8748	**29.35, 0.9249**	28.86, 0.9225
Bird	37.58, 0.9361	25.84, 0.7919	36.60, 0.9171	38.25, 0.9520	38.21, 0.9383	**39.73, 0.9559**	31.30, 0.9069

* In FFDNet, Bird performs 100 iterations, Toy performs 100 iterations. Our methods perform 100 iterations.

Table 2. The average results of PSNR in dB (left entry in each cell) and SSIM (right entry in each cell) by different algorithms on CAVE.

	2DTV	3DTV	FFDNet	DeSCI	FFDNet-TV	Ours (2DTV)	Ours (3DTV)
Average	30.70, 0.8812	30.15, 0.8906	28.65, 0.8339	31.71, 0.9153	31.26, 0.8867	34.46, 0.9318	**34.79, 0.9347**

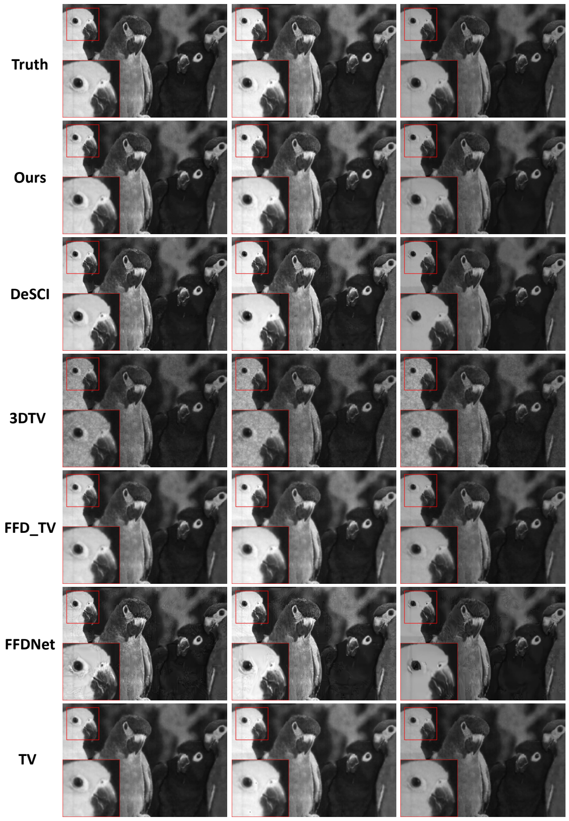

Figure 3. Simulated data: Bird. The three frames are at wavelengths 591.02nm, 630.13nm, and 674.83nm. Our results here are reconstructed by 2DTV+FFDNet.

similarity (SSIM)[7]. *The supplementary material introduces the comparison between deep learning[17] and learned prior[4] methods.* Our task used the deep network FFDNet, but it was trained on other tasks, and we directly used the network model and parameters from https://github.com/cszn/KAIR. Also, DeSCI requires GAP-TV as its initialization while ours not. We reconstruct Bird and Toy image based on the code released by [14].

While in the CAVE [27] experiment, we set the iteration of GAP-TV to 250 so that the algorithm can fully converge. Handcrafted GAP-TV iteration numbers for each data in the CAVE can obtain better results, but this process could be time-consuming. The iteration number of DeSCI is 60. We consider that the comparison between various methods is fair because we don't intentionally set the iteration number of any algorithm. In simulated data, our proposed method outperforms the previous sate-of-the-art non-deep learning method DeSCI. The performance of our method is comparable to DeSCI on real data, and more details can be recovered while saving a lot of time. Besides, we use PyTorch to implement TV denoiser so that the use of GPU can increase the speed of our algorithm.

4.1. Simulated Data

The shifting random binary mask [15] is used in our simulation. Bird and Toy data are provided by [14]. We generate a random mask for each data in CAVE.

Bird and Toy Bird[6] and Toy data are selected in the hyperspectral image reconstruction experiment of [14]. This paper also chooses these two data for the experiment. Bird consists of 24 spectral bands, and the size of each spectral band is 1021 × 703. Toy data comes from [27], which consists of 31 bands and the size of each band is 512 × 512. We use exactly the same data as [14] for the experiment. The results of Bird and Toy are tabulated in Table 1. In the table, the results of our method are presented with gray background. And 'Ours (2DTV)' in the table means that the experiment combines FFDNet and 2DTV denoiser, and 'Ours (3DTV)' means that FFDNet and 3DTV denoiser are combined. As we can see in Figure 3 and Figure 4[8], our method can recover images with more details than other methods.

CAVE To verify the effectiveness of our method, we conduct experiments on the entire CAVE dataset. CAVE includes 32 hyperspectral images, and each image contains 31 spectral bands. The image size of each band is 512 × 512. The average results of different methods applied to CAVE are in Table 2 (the results of our method are presented with

[7] We use python library scikit-image to calculate these metrics. We first clip the image into the interval $[0, 1]$. Then images are converted into unsigned integers in 0-255. Finally, performance is evaluated based on the converted images.

[8] In all figures of this paper, the results generated by our method combine FFDNet and TV priors.

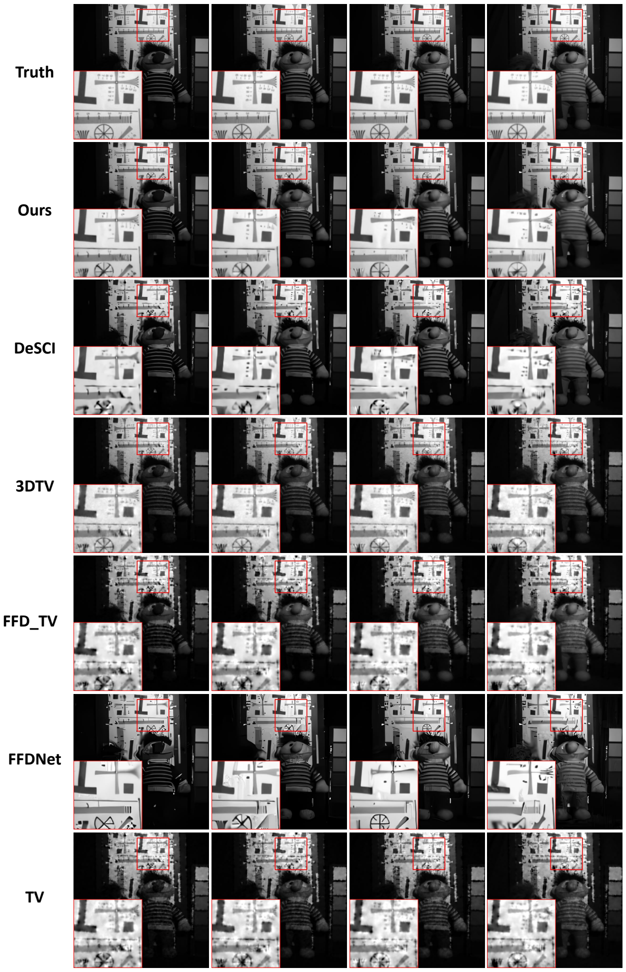

Figure 4. Simulated data: `Toy`. The four frames are at wavelengths 580nm, 620nm, 660nm, and 700nm. Our results here are reconstructed by 2DTV+FFDNet.

gray background). Our average result is higher than DeSCI about 3dB. The results of all images in CAVE is tabulated in the supplementary material. Our method can preserve more details in the reconstructed image while there are more artifacts in the reconstructed image of DeSCI (some results shown in the supplementary material).

The intensity of some simulated data is shown in the supplementary material. We randomly pick the green box area in these images to calculate the intensity.

4.2. Real Data

We obtained real data `Bird` from [14][9]. Compared with DeSCI, the image reconstructed by our method has more details, and the result of DeSCI is a bit blurry (shown in the supplementary material). In addition, our method can save a lot of time. In the experiment, our method takes about 20 minutes to reconstruct these images, while DeSCI takes about 6 hours and 20 minutes. Figure 5 shows the intensity of `Bird`. We select the same areas as [14]. The image reconstructed by our method has a larger correlation coefficient than others.

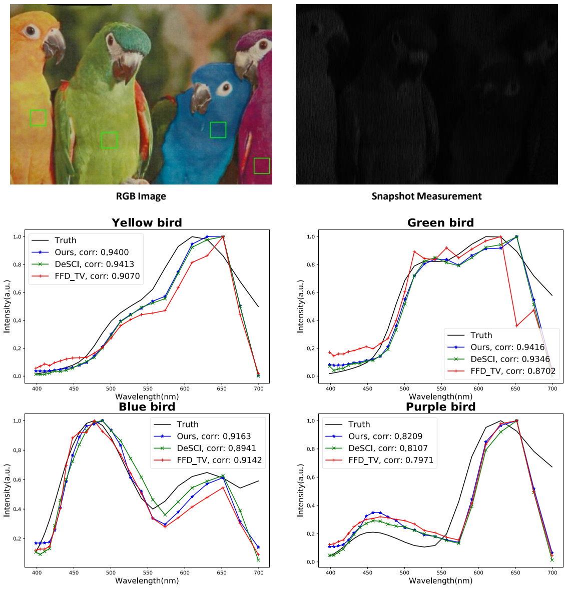

Figure 5. Real data: Spectral curves of real `Bird` hyperspectral image. The areas selected are the same as [14].

5. Conclusion

In this work, we propose a newly effective method to combine the FFDNet and TV priors to improve the existing PnP SCI algorithm for hyperspectral image reconstruction. Extensive experiments on both simulated and real datasets demonstrate that our method can take advantages of both two priors and make them mutually promote. That is to say, our method obtains better results than using FFDNet or TV alone. Also, our method is a general framework for any PnP algorithm, and thus could be extended to deal with other imaging applications.

Acknowledgments. This research was supported in part by National Key R&D Program of China (2018YFB1402600, 2020YFA0713900), the China NSFC projects (11971374, 11690011, 61721002, U1811461) and the Key Research Program of Hunan Province of China (2017GK2273).

References

[1] Emmanuel J Candès, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from

[9] This paper also provides another real data `object`. The result of `object` is reported in the supplementary material.

highly incomplete frequency information. *IEEE Transactions on information theory*, 52(2):489–509, 2006. 1

[2] Xun Cao, Tao Yue, Xing Lin, Stephen Lin, Xin Yuan, Qionghai Dai, Lawrence Carin, and David J Brady. Computational snapshot multispectral cameras: Toward dynamic capture of the spectral world. *IEEE Signal Processing Magazine*, 33(5):95–108, 2016. 1

[3] Stanley H Chan, Xiran Wang, and Omar A Elgendy. Plug-and-play admm for image restoration: Fixed-point convergence and applications. *IEEE Transactions on Computational Imaging*, 3(1):84–98, 2016. 1

[4] Inchang Choi, Daniel S. Jeon, Giljoo Nam, Diego Gutierrez, and Min H. Kim. High-quality hyperspectral reconstruction using a spectral prior. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia 2017)*, 36(6):218:1–13, 2017. 7

[5] David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006. 1, 3

[6] Michael E Gehm, Renu John, David J Brady, Rebecca M Willett, and Timothy J Schulz. Single-shot compressive spectral imaging with a dual-disperser architecture. *Optics express*, 15(21):14013–14027, 2007. 1, 7

[7] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012. 4

[8] Yasunobu Hitomi, Jinwei Gu, Mohit Gupta, Tomoo Mitsunaga, and Shree K Nayar. Video from a single coded exposure photograph using a learned over-complete dictionary. In *2011 International Conference on Computer Vision*, pages 287–294. IEEE, 2011. 1

[9] Shirin Jalali and Xin Yuan. Compressive imaging via one-shot measurements. In *2018 IEEE International Symposium on Information Theory (ISIT)*, pages 416–420. IEEE, 2018. 3

[10] Shirin Jalali and Xin Yuan. Snapshot compressed sensing: performance bounds and algorithms. *IEEE Transactions on Information Theory*, 65(12):8005–8024, 2019. 3

[11] Kyong Hwan Jin, Michael T McCann, Emmanuel Froustey, and Michael Unser. Deep convolutional neural network for inverse problems in imaging. *IEEE Transactions on Image Processing*, 26(9):4509–4522, 2017. 2

[12] Kuldeep Kulkarni, Suhas Lohit, Pavan Turaga, Ronan Kerviche, and Amit Ashok. Reconnet: Non-iterative reconstruction of images from compressively sensed measurements. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 449–458, 2016. 2

[13] Xuejun Liao, Hui Li, and Lawrence Carin. Generalized alternating projection for weighted-2,1 minimization with applications to model-based compressive sensing. *SIAM Journal on Imaging Sciences*, 7(2):797–823, 2014. 1

[14] Yang Liu, Xin Yuan, Jinli Suo, David J Brady, and Qionghai Dai. Rank minimization for snapshot compressive imaging. *IEEE transactions on pattern analysis and machine intelligence*, 41(12):2990–3006, 2018. 1, 2, 3, 7, 8

[15] Patrick Llull, Xuejun Liao, Xin Yuan, Jianbo Yang, David Kittle, Lawrence Carin, Guillermo Sapiro, and David J Brady. Coded aperture compressive temporal imaging. *Optics express*, 21(9):10526–10545, 2013. 1, 7

[16] Jiawei Ma, Xiao-Yang Liu, Zheng Shou, and Xin Yuan. Deep tensor admm-net for snapshot compressive imaging. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 10223–10232, 2019. 2

[17] Xin Miao, Xin Yuan, Yunchen Pu, and Vassilis Athitsos. lambda-net: Reconstruct hyperspectral images from a snapshot measurement. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4058–4068. IEEE, 2019. 2, 7

[18] Ali Mousavi and Richard G Baraniuk. Learning to invert: Signal recovery via deep convolutional networks. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 2272–2276. IEEE, 2017. 2

[19] Dikpal Reddy, Ashok Veeraraghavan, and Rama Chellappa. P2c2: Programmable pixel compressive camera for high speed imaging. In *CVPR 2011*, pages 329–336. IEEE, 2011. 1

[20] Ernest K Ryu, Jialin Liu, Sicheng Wang, Xiaohan Chen, Zhangyang Wang, and Wotao Yin. Plug-and-play methods provably converge with properly trained denoisers. *arXiv preprint arXiv:1905.05406*, 2019. 2, 5, 6

[21] Ayan Sinha, Justin Lee, Shuai Li, and George Barbastathis. Lensless computational imaging through deep learning. *Optica*, 4(9):1117–1125, 2017. 2

[22] Yangyang Sun, Xin Yuan, and Shuo Pang. High-speed compressive range imaging based on active illumination. *Optics express*, 24(20):22836–22846, 2016. 1

[23] Yangyang Sun, Xin Yuan, and Shuo Pang. Compressive high-speed stereo imaging. *Optics express*, 25(15):18182–18190, 2017. 1

[24] Ashwin Wagadarikar, Renu John, Rebecca Willett, and David Brady. Single disperser design for coded aperture snapshot spectral imaging. *Applied optics*, 47(10):B44–B51, 2008. 1

[25] Ashwin A Wagadarikar, Nikos P Pitsianis, Xiaobai Sun, and David J Brady. Video rate spectral imaging using a coded aperture snapshot spectral imager. *Optics express*, 17(8):6368–6388, 2009. 1

[26] Jianbo Yang, Xuejun Liao, Xin Yuan, Patrick Llull, David J Brady, Guillermo Sapiro, and Lawrence Carin. Compressive sensing by learning a gaussian mixture model from measurements. *IEEE Transactions on Image Processing*, 24(1):106–119, 2014. 2

[27] F. Yasuma, T. Mitsunaga, D. Iso, and S.K. Nayar. Generalized Assorted Pixel Camera: Post-Capture Control of Resolution, Dynamic Range and Spectrum. Technical report, Nov 2008. 7

[28] Xin Yuan. Generalized alternating projection based total variation minimization for compressive sensing. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 2539–2543. IEEE, 2016. 1, 2

[29] Xin Yuan and Raziel Haimi-Cohen. Image compression based on compressive sensing: End-to-end comparison with jpeg. *IEEE Transactions on Multimedia*, 2020. 2

[30] Xin Yuan, Hong Jiang, Gang Huang, and Paul A Wilford. Compressive sensing via low-rank gaussian mixture models. *arXiv preprint arXiv:1508.06901*, 2015. 1, 2

[31] Xin Yuan, Yang Liu, Jinli Suo, and Qionghai Dai. Plug-and-play algorithms for large-scale snapshot compressive imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1447–1457, 2020. 1, 2, 3, 5, 6

[32] Xin Yuan, Patrick Llull, Xuejun Liao, Jianbo Yang, David J Brady, Guillermo Sapiro, and Lawrence Carin. Low-cost compressive sensing for color video and depth. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3318–3325, 2014. 1

[33] Zhiyuan Zha, Xin Yuan, Bihan Wen, Jiantao Zhou, Jiachao Zhang, and Ce Zhu. From rank estimation to rank approximation: Rank residual constraint for image restoration. *IEEE Transactions on Image Processing*, 29:3254–3269, 2019. 2

[34] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing*, 27(9):4608–4622, 2018. 1, 2