Ensemble Learning Priors Driven Deep Unfolding for Scalable Video Snapshot Compressive Imaging

Chengshuai Yang[0000−0003−2840−5344], Shiyu Zhang[0000−0001−7111−3895], and Xin Yuan✉[0000−0002−8311−7524]

School of Engineering, Westlake University, Hangzhou, Zhejiang 310030, China
integrityyang@gmail.com {zhangshiyu, xyuan}@westlake.edu.cn

Abstract. Snapshot compressive imaging (SCI) can record a 3D datacube by a 2D measurement and algorithmically reconstruct the desired 3D information from that 2D measurement. The reconstruction algorithm thus plays a vital role in SCI. Recently, deep learning (DL) has demonstrated outstanding performance in reconstruction, leading to better results than conventional optimization-based methods. Therefore, it is desirable to improve DL reconstruction performance for SCI. Existing DL algorithms are limited by two bottlenecks: 1) a high-accuracy network is usually large and requires a long running time; 2) DL algorithms are limited by scalability, *i.e.*, a well-trained network cannot generally be applied to new systems. To this end, this paper proposes to use **ensemble learning priors** in DL to achieve high reconstruction speed and accuracy in a single network. Furthermore, we develop the scalable learning approach during training to empower DL to handle data of different sizes without additional training. Extensive results on both simulation and real datasets demonstrate the superiority of our proposed algorithm. The code and model can be accessed at https://github.com/integritynoble/ELP-Unfolding/tree/master.

Keywords: Deep Unfolding, Ensemble, Snapshot Compressive Imaging, Scalable Learning

1 Introduction

Recently, video snapshot compressive imaging (SCI) [7, 30, 60] has attracted much attention because it can improve imaging speed by capturing three-dimensional (3D) information from 2D measurement. When video SCI works, multiple frames are first modulated by different masks (in the optical domain), and these modulated frames are mapped into a single measurement. After this, the reconstruction algorithm recovers these multiple frames from single measurement [56]. At present, the mask can easily be adjusted with a higher speed than the capture rate of the camera [20, 38, 41]. Thus, SCI enjoys the advantages of high speed, low memory, low bandwidth, low power and potentially low cost [58, 59].

✉ Corresponding author.

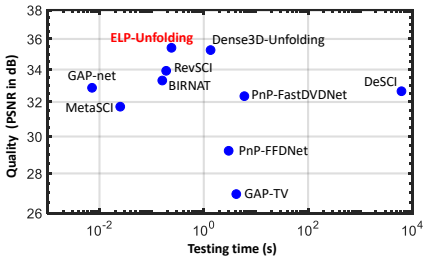

Fig. 1: Trade-off between quality and testing-time of various algorithms for SCI reconstruction. Our proposed Ensemble Learning Priors (ELP) unfolding achieves the state-of-the-art results in a short testing time. Besides, after scalable learning, our ELP-Unfolding can be used in different masks and different compression ratios and thus can be applied to various scenes by a single trained model.

How to recover the original multiple frames from the single measurement always plays a vital role in SCI. Recently, deep learning reconstruction methods have outperformed traditional iterative reconstruction methods not only in reconstruction accuracy but also in test time [8–10,24,34,43,50,51]. But most deep learning methods lack interpretability. To increase interpretability, deep unfolding method has been developed, which simulates the iterative algorithm [17, 31, 51, 63]. Deep unfolding method adopts iterative framework but replaces traditional denoiser (such as total variation [4, 26] and nonlocal self-similarity [12, 29]) with the trained neural network denoiser. So far, the deep unfolding method has achieved the best result for SCI. Among deep unfolding algorithms, GAP-net [31] can use the shortest time (0.0072 s) to achieve 32 dB for PSNR for benchmark dataset. Dense3D-Unfolding [46] achieved the best result (35 dB), though it costs a long time (1.35 s) due to the use of complex 3D convolutional neural networks (CNNs). Thus, *the speed and accuracy have not coexisted in one algorithm yet*. What's worse, most of these deep learning algorithms are limited by scalability. To apply the trained model to new systems, the model usually should be trained again. Although MetaSCI [45] can be quickly applied to new SCI modulations (in spatial but not in temporal dimension), it still requires adaptation (retraining).

Bearing the above concerns in mind, in order to achieve a higher reconstruction accuracy with a high computing speed, we develop the Ensemble Learning Priors (ELP) unfolding based on 2D-CNN for SCI. Specifically, 2D-CNN can retain fast processing and ensemble learning can increase reconstruction accuracy. Ensemble learning is powerful in achieving reconstruction accuracy and has also achieved state-of-the-art (SOTA) results in a number of models on other tasks [36, 65, 68], due to the fact that mutiple models/priors have complementary advantages over a single model/prior. Fortunately, the deep unfolding algorithm can include many neural network priors, even if these priors stay at different (iteration) stages. In this paper, we first propose to gather multiple neural network priors in one stage to realize **ensemble learning** for SCI without increasing training time. To further increase the reconstruction accuracy, dense connection is employed in our network, which can help the latter (stage) models learn some useful information from the previous (stage) models. In this manner, our ELP-unfolding can achieve SOTA result, outperform Dense3D-Unfolding [46], and use a shorter running time.

Furthermore, to realize the scalability, we develop a scalable learning procedure for SCI. Our method *not only has scalability in the spatial dimensions but also in the temporal dimension*. Considering the spatial scalability, we set our ELP-unfolding to be fully convolutional without the multilayer perception (MLP) structure. For temporal dimension scalability, the input of neural network priors is set to have the same channels even for different temporal dimensional scenes. Based on this, our scalable learning method can have the same capability as traditional iteration algorithms, to be applied to different systems. Specific contributions of our paper are listed as follows:

- We develop the **ensemble learning prior unfolding** for SCI. ELP unfolding is a general method for inverse algorithms, which can also be applied to other fields, such as single pixel camera [16, 19, 40], MRI [2, 28], lensless imaging [3, 57, 61], spectral compressive imaging [6, 18, 21, 27, 32, 66], and tomography imaging [11, 39, 52].
- We first propose the **scalable learning for SCI**. After training once, our model can be used in new systems with different modulations or different compression ratios. Besides, scalable learning can achieve better results than PnP algorithm with a fast inference speed.
- We adopt skip connection techniques in unfolding. In our ELP-unfolding, the skip connection only uses the simple adding and concatenating. By contrast, the Dense3D-Unfolding [46] adopts complex methods such as DFMA (dense feature map adaption) to realize connection.
- Our method **achieves SOTA results** for SCI in benchmark dataset based on 2D-CNN, outperforming the 3D-CNN method at a faster inference speed [46] as shown in Fig. 1.

In a nutshell, our ensemble learning priors unfolding has two periods. In the first period, i.e. a single prior period, each stage contains one neural network prior. Afterwards, in the second (ensemble priors) period, each stage contains all previous stage priors in the ensemble manner.

2 Related Work

SCI is related to compressive sensing (CS) [22, 67], where reconstruction is significantly important as it provides the desired signals (such as images) from the compressed measurements. For CS [22, 24, 31, 34, 43], there are two kinds of reconstruction methods: traditional iterative method and deep learning method. The traditional iteration method contains a lot of iterations and each iteration contains the projection operation and denoising operation (and optionally some other steps). The denoising operation generally determines the performance of one algorithm. For example, total variation [4, 26] denoiser has a fast speed but usually can only provide blurry images while the nonlocal self-similarity based denoiser [12, 29] can achieve a clearer image but take a long time. Recently, deep learning has shown strong power in reconstructing images [24, 34, 43, 50, 51]. At first, deep learning was regarded as a black box and the trained model can

get the better images than traditional iterative method at a fast speed. As a black box to train, the trained model will contain measurement matrix (masks) information. Thus, the training model usually can not be applied to new masks (such as a new hardware system). To address this problem, the deep unfolding method for CS has be developed. Deep unfolding method simulates traditional iterative method using a few iterations (stages), each of which has projection operation and denoising operation. Different from traditional methods, deep unfolding uses a trained neural network as a denoising prior. Therefore, deep learning mainly contributes to denoising in deep unfolding method with little dependence on mask information. The mask information is mainly processed by the projection operation. Thus, deep unfolding algorithms has a strong robustness to a variety of masks [31,63]. Besides, Dense3D-Unfolding [46] obtained SOTA for SCI by combining deep unfolding method and 3D-CNN, but at the cost of slow computation. Though the unfolding method can solve the scalability problem of various masks, the scalability problem of various sizes (both spatial size and temporal size, a.k.a., the compression ratio) still remains in unfolding method. Deep unfolding method still does not have the same scalability as the traditional iterative method.

To address these challenges, in this paper, we develop the ensemble learning priors unfolding for scalable SCI. We use ensemble learning and 2D-CNN to realize high reconstruction accuracy and speed, and develop scalable learning to realize scalability.

3 Preliminary: Video SCI System

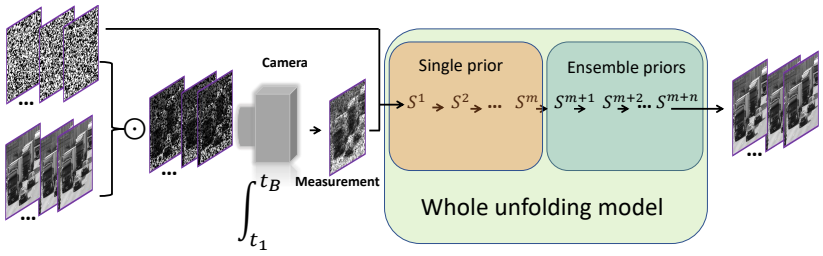

Fig. 2: Principle of Video SCI (left) and our ELP-unfolding (right). **Left**: the high speed dynamic scene at timestamps t_1 to t_B, encoded by high-speed variant masks (dynamic coded apertures) and then integrated to a single coded measurement (a compressed image) $\mathbf{Y}$. **Right**: our whole ELP-unfolding reconstructs the original dynamic scene from the masks $\{\mathbf{C}_1, \ldots, \mathbf{C}_B\}$ and the compressed image $\mathbf{Y}$, which includes the single prior period in Fig. 3(a) and ensemble priors period in Fig. 3(b). S^m represents the m^{th} stage.

As depicted in Fig. 2, let $\{\mathbf{X}_1, \ldots, \mathbf{X}_B\}$ denote the discretized video frames at timestamps $\{t_1, \ldots, t_B\}$. These video frames are modulated by dynamic coded aperture, a.k.a., the masks $\{\mathbf{C}_1, \ldots, \mathbf{C}_B\}$, respectively. The modulated frames are

then integrated into a single coded measurement (a compressed image) $\mathbf{Y}$. Here, $\{\mathbf{X}_b\}_{b=1}^B \in \mathbb{R}^{n_x \times n_y \times B}$, $\{\mathbf{C}_b\}_{b=1}^B \in \mathbb{R}^{n_x \times n_y \times B}$ and $\mathbf{Y} \in \mathbb{R}^{n_x \times n_y}$. This forward model can be written as

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

where $\odot$ and $\mathbf{Z} \in \mathbb{R}^{n_x \times n_y}$ denote the matrix element-wise product and noise, respectively. Eq. (1) is equivalent to the following linear form

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

where $\boldsymbol{y} = \text{Vec}(\mathbf{Y}) \in \mathbb{R}^{n_x n_y}$,$\boldsymbol{z} = \text{Vec}(\mathbf{Z}) \in \mathbb{R}^{n_x n_y}$ and $\boldsymbol{x} = \text{Vec}(\mathbf{X}) = [\text{Vec}(\mathbf{X}_1), \ldots, \text{Vec}(\mathbf{X}_B)] \in \mathbb{R}^{n_x n_y B}$. Different from traditional compressive sensing [13–15], the sensing matrix $\mathbf{H}$ in (2) has a very special structure and can be written as

$$\mathbf{H} = [\mathbf{D}_1, \ldots, \mathbf{D}_B], \tag{3}$$

where $\{\mathbf{D}_b = diag(\text{Vec}(\mathbf{C}_b)) \in \mathbb{R}^{n_x n_y \times n_x n_y}\}_{b=1}^B$are diagonal matrices of masks. Therefore, the compressive sampling rate in SCI is equal to $1/B$. The reconstruction error of SCI is bounded even when $B > 1$ [22].

4 Our proposed methods

4.1 Ensemble learning priors unfolding for SCI

Given the compressed measurement $\mathbf{Y}$ and coding pattern $\{\mathbf{C}_b\}_{b=1}^B$ captured by the SCI system, there exist two optimization frameworks to predict the desired high speed frames $\{\mathbf{X}_b\}_{b=1}^B$: penalty function method and augmented Lagrangian (AL) method. The performance of AL method is better than that of the penalty function method, which has been proved in previous work [1, 25, 48]. Therefore the AL method is adopted here, which is formulated as follows:

$$\boldsymbol{x} = \text{argmin}_{\boldsymbol{x}} \, \Phi(\boldsymbol{x}) - \boldsymbol{\lambda}_1^T(\boldsymbol{y} - \mathbf{H}\boldsymbol{x}) + \tfrac{\gamma_1}{2} \left\|\boldsymbol{y} - \mathbf{H}\boldsymbol{x}\right\|_2^2, \tag{4}$$

where $\Phi(\boldsymbol{x}), \boldsymbol{\lambda}_1$ and γ_1 denote the prior regularization, Lagrangian multiplier and penalty parameter, respectively. For convenience, Eq. (4) is further written as

$$\boldsymbol{x} = \text{argmin}_{\boldsymbol{x}} \, \Phi(\boldsymbol{x}) + \tfrac{\gamma_1}{2} \left\|\boldsymbol{y} - \mathbf{H}\boldsymbol{x} - \tfrac{\boldsymbol{\lambda}_1}{\gamma_1}\right\|_2^2. \tag{5}$$

Single prior. To solve Eq. (5), an auxiliary variable $\boldsymbol{v}$ is introduced. Then Eq. (5) is further written as

$$\boldsymbol{x} = \text{argmin}_{\boldsymbol{x}} \, \Phi(\boldsymbol{v}) + \tfrac{\gamma_1}{2} \left\|\boldsymbol{y} - \mathbf{H}\boldsymbol{x} - \tfrac{\boldsymbol{\lambda}_1}{\gamma_1}\right\|_2^2 \text{ subject to } \boldsymbol{v} = \boldsymbol{x}. \tag{6}$$

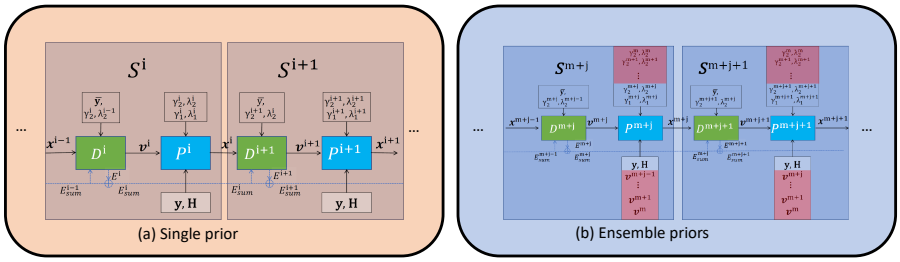

Fig. 3: (a) Principle of the single prior period. Here, D^i represents the i^{th} denoising operation, as in Eq. (11) while P^i represents the i^{th} projection operation, as in Eq. (10). (b) Principle of ensemble priors period. Here, several denoising results $\boldsymbol{v}^m ... \boldsymbol{v}^{m+1}$ are gathered together projection operation.

By adopting alternating direction method of multipliers (ADMM) method [5, 49], Eq. (6) is further written as

$$\boldsymbol{x}, \boldsymbol{v} = \text{argmin}_{\boldsymbol{x},\boldsymbol{v}}\, \Phi(\boldsymbol{v}) + \frac{\gamma_2}{2}\left\|\boldsymbol{x} - \boldsymbol{v} - \frac{\boldsymbol{\lambda}_2}{\gamma_2}\right\|_2^2 + \frac{\gamma_1}{2}\left\|\boldsymbol{y} - \mathbf{H}\boldsymbol{x} - \frac{\boldsymbol{\lambda}_1}{\gamma_1}\right\|_2^2. \tag{7}$$

According to ADMM, Eq. (7) can be divided into the two subproblems and solved iteratively, as shown in Fig. 3(a)

$$\boldsymbol{v}^i = \text{argmin}_{\boldsymbol{v}}\, \Phi(\boldsymbol{v}) + \frac{\gamma_2^i}{2}\left\|\boldsymbol{x}^{i-1} - \boldsymbol{v} - \frac{\boldsymbol{\lambda}_2^i}{\gamma_2^i}\right\|_2^2, \tag{8}$$

$$\boldsymbol{x}^i = \text{argmin}_{\boldsymbol{x}}\, \frac{\gamma_2^i}{2}\left\|\boldsymbol{x} - \boldsymbol{v}^i - \frac{\boldsymbol{\lambda}_2^i}{\gamma_2^i}\right\|_2^2 + \frac{\gamma_1^i}{2}\left\|\boldsymbol{y} - \mathbf{H}\boldsymbol{x} - \frac{\boldsymbol{\lambda}_1^i}{\gamma_1^i}\right\|_2^2, \tag{9}$$

where the superscript i denotes the iteration index.

For subproblem $\boldsymbol{x}_i$, there exists a closed-form solution, which is called projection operation

$$\boldsymbol{x}^i = (\gamma_2^i\mathbf{I} + \gamma_1^i\mathbf{H}^T\mathbf{H})^{-1}\left[\boldsymbol{\lambda}_2^i + \gamma_2^i\boldsymbol{v}^i + \mathbf{H}^T\gamma_1^i(\boldsymbol{y} - \frac{\boldsymbol{\lambda}_1^i}{\gamma_1^i})\right]. \tag{10}$$

Due to the special structure of $\mathbf{H}$, this can be solved in one shot [29].

For subproblem $\boldsymbol{v}_i$, Eq. (8) can be rewritten as

$$\boldsymbol{v}^i = \text{argmin}_{\boldsymbol{v}}\, \Phi(\boldsymbol{v}) + \frac{\gamma_2^i}{2}\left\|\boldsymbol{u}^{i-1} - \boldsymbol{v}\right\|_2^2, \tag{11}$$

where $\boldsymbol{u}^{i-1} = \boldsymbol{x}^{i-1} - \frac{\boldsymbol{\lambda}_2^{i-1}}{\gamma_2^i}$. Eq. (11) is a classical denoising problem, which can be solved by denoising prior such as TV, wavelet transformation, denoising network, *etc.*In this paper, denoising network prior is adopted as shown in Fig. 4.

Ensemble priors. In every stage of unfolding, the denoising prior has different parameters and thus plays different roles in removing noise, even these priors have the same structure. To take full use of different denoisers among different stages, these priors after $\boldsymbol{m}$ stages are gathered together to perform projection

operation to produce $\boldsymbol{x}$. Therefore, Eq. (8) and Eq. (9) in ensemble priors period, as shown in Fig. 3(b), becomes

$$\boldsymbol{v}^{m+j} = \text{argmin}_{\boldsymbol{v}}\, \Phi(\boldsymbol{v}_1) + \frac{\gamma_2^{m+j}}{2} \left\| \boldsymbol{x}^{m+j-1} - \boldsymbol{v} - \frac{\boldsymbol{\lambda}_2^{m+j-1}}{\gamma_2^{m+j}} \right\|_2^2, \tag{12}$$

and

$$\begin{aligned} \boldsymbol{x}^{m+j} = &\text{argmin}_{\boldsymbol{x}}\, \frac{\gamma_2^m}{2} \left\| \boldsymbol{x} - \boldsymbol{v}^m - \frac{\boldsymbol{\lambda}_2^m}{\gamma_2^m} \right\|_2^2 + \frac{\gamma_2^{m+1}}{2} \left\| \boldsymbol{x} - \boldsymbol{v}^{m+1} - \frac{\boldsymbol{\lambda}_2^{m+1}}{\gamma_2^{m+1}} \right\|_2^2 + \cdots \\ &+ \frac{\gamma_1^{m+j}}{2} \left\| \boldsymbol{y} - \mathbf{H}\boldsymbol{x} - \frac{\boldsymbol{\lambda}_1^{m+j}}{\gamma_1^{m+j}} \right\|_2^2. \end{aligned} \tag{13}$$

For subproblem $\boldsymbol{v}_i$, Eq. (12) can still adopt the same denoising prior form as in single-prior period. For subproblem $\boldsymbol{x}_i$, there is a slightly difference because of ensemble

$$\begin{aligned} \boldsymbol{x}^{m+j} = &[(\gamma_2^m + \gamma_2^{m+1} + \cdots + \gamma_2^{m+j})\mathbf{I} + \mathbf{H}^T \gamma_1^{m+j} \mathbf{H}]^{-1} \\ &\left[\boldsymbol{\lambda}_2^m + \gamma_2^m \boldsymbol{v}^m + \cdots + \boldsymbol{\lambda}_2^{m+j} + \gamma_2^{m+j} \boldsymbol{v}^{m+j} + \mathbf{H}^T \gamma_1^{m+j} (\boldsymbol{y} - \frac{\boldsymbol{\lambda}_1^{m+j}}{\gamma_1^{m+j}}) \right]. \end{aligned} \tag{14}$$

Last but not least, the Lagrangian multipliers $\boldsymbol{\lambda}_1^i$ and $\boldsymbol{\lambda}_2^i$ are updated by

$$\boldsymbol{\lambda}_1^i = \boldsymbol{\lambda}_1^{i-1} - \gamma_1^i (\boldsymbol{y} - \mathbf{H}\boldsymbol{x}^{i-1}), \tag{15}$$

$$\boldsymbol{\lambda}_2^i = \boldsymbol{\lambda}_2^{i-1} - \gamma_2^i (\boldsymbol{x} - \boldsymbol{v}^{i-1}). \tag{16}$$

Besides, the γ_1^i and γ_2^i are trained with the denoising prior parameters at every stage.

In our method, the whole algorithm body should be divided into two parts: a single prior period and an ensemble priors period, because the first several stages can only provide rough estimates. If the priors in the first several stages are coupled to the latter stages, the poor performance of the first several priors will worsen the whole algorithm performance. There are 13 stages in our algorithm, the first 8 stages are single prior periods and latter 5 stages are ensemble priors periods. It is noted that there are 6 priors in last stage. As we can see in Eq. (13) and Eq. (14), there exist six $\boldsymbol{v}$'s if $j = 5$.

Denoising prior structure. As shown in Fig. 4, U-net [42] is used as the backbone for denoising prior, which we adopt from FastDVDnet [44], but here we remove batch normalization and quadruple the depth; this means that the channels for three different features are 128, 256 and 512, respectively. Thus, the training parameters of our proposed ELP-unfolding mainly consists of these 13 U-net structures. More details can be found in the supplementary materials (SM). Following [44, 64], the penalty parameter γ_2^i is expanded to a noise map as part of the input. To help denoising, the normalized measurement $\overline{\mathbf{Y}}$ is also added to the input [8, 45, 46], which is defined as

$$\overline{\mathbf{Y}} = \mathbf{Y} \oslash \textstyle\sum_{b=1}^{B} \mathbf{C}_b, \tag{17}$$

where $\oslash$ represents the matrix element-wise division. Therefore, the input consists of noise map γ_2^i, normalized measurement $\overline{\mathbf{Y}}$ and $\boldsymbol{x}^{i-1} - \frac{\boldsymbol{\lambda}_2^{i-1}}{\gamma_2^i}$, and the output is $\boldsymbol{v}^i$. Besides, dense connection is employed in the denoising prior network design.

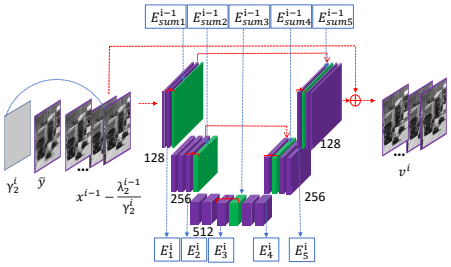

Fig. 4: Denoising prior structure based on U-net [42]. To realize connection, sum feature $\mathbf{E}_{sumj}^{i-1}$ from previous priors is coupled into current prior and current feature $\mathbf{E}_j^i$ is used to help form next sum feature $\mathbf{E}_{sumj}^i$.

Algorithm 1 ELP-unfolding for SCI Reconstruction

Require: $\mathbf{H}$, $\boldsymbol{y}$, $\overline{\mathbf{Y}}$, $\{\gamma_1^0, \ldots, \gamma_1^{m+n}\}$, $\{\gamma_2^0, \ldots, \gamma_2^{m+n}\}$.

Initial $\boldsymbol{v}^0 = \mathbf{0}$, $\boldsymbol{\lambda}_1^0 = \mathbf{0}$, $\boldsymbol{\lambda}_2^0 = \mathbf{0}$.
Update $\boldsymbol{x}^0$ by Eq. (10)
% single prior period
for i = 1, ... , m **do**
  Update $\boldsymbol{v}^i$ by Eq. (11), $\boldsymbol{\lambda}_2^i$ by Eq. (16).
  Update $\boldsymbol{\lambda}_1^i$ by Eq. (15), $\boldsymbol{x}^i$ by Eq. (10).
end for
% ensemble priors period
for k = m+1, ... , m+n **do**
  Update $\boldsymbol{v}^k$ by Eq. (11), $\boldsymbol{\lambda}_2^k$ by Eq. (16).
  Update $\boldsymbol{\lambda}_1^k$ by Eq. (15), $\boldsymbol{x}^k$ by Eq. (14).
end for

Dense connection for unfolding. In traditional unfolding method, the connection between two stages are $\boldsymbol{v}$ and $\boldsymbol{u}$, that is $\boldsymbol{x}^{i-1} - \frac{\boldsymbol{\lambda}_2^{i-1}}{\gamma_2^i}$, which have a small number of temporal dimensions. Therefore, most latent information in U-net structure cannot be transferred between different priors. To break this bottleneck, the skip connection technique is used here. As shown in Fig. 4, in the $\boldsymbol{i}^{th}$ prior, the feature $\mathbf{E}_j^i$ and feature $\mathbf{E}_{sumj}^{i-1}$ operate in the latent space of U-net structure as a whole feature. Besides, the feature $\mathbf{E}_j^i$ will add to $\mathbf{E}_{sumj}^{i-1}$ to form $\mathbf{E}_{sumj}^i$, that is, $\mathbf{E}_{sumj}^i = \mathbf{E}_{sumj}^{i-1} + \mathbf{E}_j^i$.

By re-ordering the updating equations, we summarize the entire algorithm in Algorithm 1.

4.2 Scalable learning for SCI

Existing deep learning methods usually have limited scalability, *i.e.*, one trained model can only be applied to one system with specific masks and compression ratio B. When the scene data size changes, the new corresponding model usually needs to be trained again. The most recent MetaSCI [45] can quickly be applied to a new model but also demands new adaptation process. In addition, MetaSCI adaptation is limited in space but not suitable for time (compression ratio). Even some deep learning methods that are independent of multi-layer perception, such as Dense3D-Unfolding, can be applied to different spatial size cases, but they have no temporal scalability. They must be trained again for new applications with different temporal dimensions B.

To address this problem, we develop scalable learning for SCI. This scalable learning has scalability not only in the spatial dimension but also in the temporal dimension. Specifically, to ensure spatial scalability, we only employ the convolutional

Fig. 5: Selected reconstruction results of benchmark dataset by GAP-TV [55], DeSCI [29], PnP-FFDNet [58], RevSCI [8] and the proposed ELP-unfolding (Please zoom-in to see details).

neural network, ignoring MLP; to ensure temporal scalability, we train a scalable frames model within a certain number of frames, which is the maximum frames M. During training, the number of frames (smaller than M) is randomly chosen; M is also the number of channel in denoising networks. In most cases, the original data should be repeatedly rearranged several times to satisfy the frames number M. When M is not an integer multiple of the frame number of dynamic scene, only the first several frames of the original data are used in the last arranging process.

Even though the maximum temporal size needs to be pre-set, the new maximum temporal model can conveniently use the previous different maximum temporal models as the pre-trained model to speed up the training process.

4.3 Training

Given the measurement $\mathbf{Y}$ and masks $\{\mathbf{C}_b\}_{b=1}^{B}$, our ELP-unfolding can generate $\{\hat{\mathbf{X}}_b\}_{b=1}^{B} \in \mathbb{R}^{n_x \times n_y \times B}$. The mean square error (MSE) is selected as our loss function, expressed as

$$\ell_{MSE} = \frac{1}{SBn_x n_y} \sum_{s=1}^{S} \sum_{b=1}^{B} \left\| \mathbf{X}_b - \hat{\mathbf{X}}_b \right\|_2^2, \tag{18}$$

where $\mathbf{X}_b$ is ground truth and S is batchsize.

We use PyTorch [35] to train our model on an NVIDIA A40 GPU. For all training processes, we adopt the Adam optimizer [23] with a mini-batch size of 3 and a spatial size of 256×256. We also adopt a pre-training strategy. The whole training process has two periods. Firstly, 8 stages with a single prior model are trained as pretrained parameters. Secondly, the whole ELP-unfolding with

Table 1: Benchmark datasets: the average results of PSNR in dB (left entry in each cell) and SSIM (right entry in each cell) and run time per measurement in seconds by different algorithms on 6 benchmark datasets.

Algorithm	Kobe	Traffic	Runner	Drop	Crash	Aerial	Average	Run time (s)
GAP-TV [55]	26.92, 0.838	20.66, 0.691	29.81, 0.895	34.95, 0.966	24.48, 0.799	24.81, 0.811	26.94, 0.833	4.2 (CPU)
DeSCI [29]	33.25, 0.952	28.71, 0.925	38.48, 0.969	43.10, 0.992	27.04, 0.909	25.33, 0.860	32.65, 0.934	6180 (CPU)
PnP-FFDNet [58]	30.33, 0.925	24.01, 0.835	32.44, 0.931	39.68, 0.986	24.67, 0.833	24.29, 0.820	29.21, 0.888	3.0 (GPU)
PnP-FastDVDnet [59]	32.73, 0.947	27.95, 0.932	36.29, 0.962	41.82, 0.989	27.32, 0.925	27.98, 0.897	32.35, 0.942	6 (GPU)
BIRNAT [10]	32.71, 0.950	29.33, 0.942	38.70, 0.976	42.28, 0.992	27.84, 0.927	28.99, 0.927	33.31, 0.951	0.16 (GPU)
GAP-Unet-S12 [31]	32.09, 0.944	28.19, 0.929	38.12, 0.975	42.02, 0.992	27.83, 0.931	28.88, 0.914	32.86, 0.947	**0.0072 (GPU)**
Meta-SCI [45]	30.12, 0.907	26.95, 0.888	37.02, 0.967	40.61, 0.985	27.33, 0.906	28.31, 0.904	31.72, 0.926	0.025 (GPU)
RevSCI [8]	33.72, 0.957	30.02, 0.949	39.40, 0.977	42.93, 0.992	28.12, 0.937	29.35, 0.924	33.92, 0.956	0.19 (GPU)
Dense3D-Unfolding [46]	**35.00**, **0.969**	**31.76**, **0.966**	40.03, 0.980	44.96, **0.995**	29.33, 0.956	30.46, 0.943	35.26, 0.968	1.35 (GPU)
ELP-Unfolding (Ours)	34.41, 0.966	31.58, 0.962	**41.16**, **0.986**	**44.99**, **0.995**	**29.65**, **0.960**	**30.68**, **0.943**	**35.41**, **0.969**	0.24 (GPU)

the pretrained parameters, is then trained, with 13 stages, 6 ensemble-priors in the last stage. And the first 8 stages just contains a single prior in each stage. Besides, the former 8 stages in the entire ELP-unfolding match the pretrained model very well, completely adopting the pretrained parameters. The latter 5 stages priors adopt the same last stage parameters in the pretrained model.

Regarding the learning rate, we adopt the same strategy for these two training periods. The difference lies in the initial learning rate. For the first (pretrained) period, the initial learning rate is set to 1×10^{-4}. For the second (ELP-unfolding) period, the initial learning rate is set to 2×10^{-5}. After the first five epochs, the learning rate decays a factor of 0.9 every 15 epochs. Besides, for the first (pretrained) period, the total number of epoch is 200 and training time is about 8 days. For the second period, the total number of epoch is 320 and training time is about 13 days.

In this paper we used above training strategies to train three models, namely benchmark model, scalable model and real data model.

5 Experiment

5.1 Training Dataset

We used DAVIS2017 [37] dataset with a resolution of 480×894 (480p) as our training dataset for all experiments. Video clips with spatial size of 256 × 256 are randomly cropped from this training dataset.

5.2 Benchmark datasets for SCI

Kobe, Traffic, Runner, Drop, Crash, and Aerial are the Benchmark datasets for SCI[59], where the data-size is 256×256×8, *i.e.*n_x=n_y=256, B=8. Based on these datasets, we compare our ELP-unfolding with a special temporal size of 8 to other SOTA algorithms, including GAP-TV [55], DeSCI [29], PnP-FFDNet [58], PnP-FastDVDnet [59], BIRNAT [10], GAP-Unet-S12 [31], Meta-SCI [45], RevSCI [8], Dense3D-Unfolding [46]. The results are summarized in Table 1. As we can see, iterative algorithms including PnP based algorithms (GAP-TV, DeSCI, PnP-FFDNet, PnP-FastDVDnet) provide inferior results at a slow speed (more than

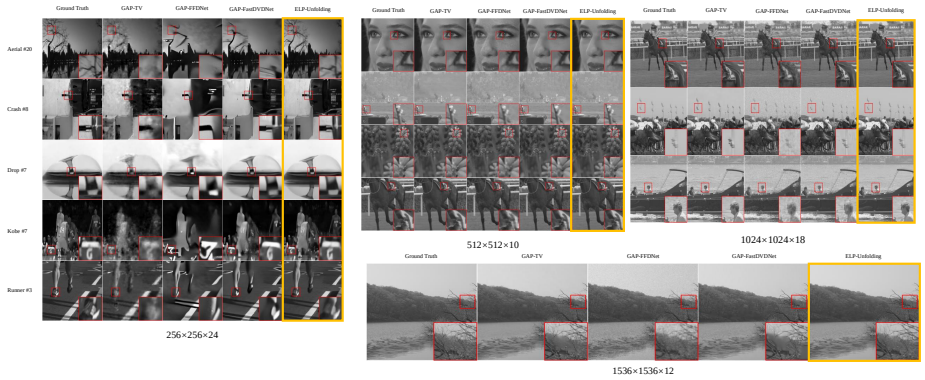

Fig. 6: Scalability: Selected results by GAP-TV, PnP-FFDNet, PnP-FastDVDnet and our ELP-unfolding with various spatial sizes and compression ratios.

one second). Deep learning algorithms can achieve better result in a short running time (usually less than 1 second).

For direct comparison of deep learning algorithms, Table 2 shows the results of top three algorithms, namely, RevSCI, Dense3D-Unfolding and ours. Although Dense3D-Unfolding has achieved the best results before, it costs a long time to test (1.35 s). Our ELP-unfolding algorithm not only achieves better result than Dense3D-Unfolding, but also saves test time (costing 0.24 s). For visualization purpose, we also present some images in Fig. 5, from the zoom areas we can see that our ELP-unfolding provides much clearer images with sharper edges and more abundant details than other algorithms, even the Dense3D-Unfolding (Crash). We also believe that by adopting 3D-CCN, ELP-unfolding can achieve even better results.

Table 2: The comparison of top three algorithms: time, memory for training one batch and reconstruction accuracy (PSNR).

	Time	Memory	PSNR
RevSCI	0.19 s	Flexible	33.92 dB
Dense3D-Unfolding	1.35 s	28.7 G	35.26 dB
Our method	0.24 s	12.5 G	35.41 dB

5.3 Scalable datasets for SCI

To verify the scalability of our ELP-unfolding method, we trained one model to test four different size datasets: 256×256×24, 512×512 ×10, 1024 ×1024×18 and 1536×1536×12. The latter three datasets are cropped from the Ultra Video Group (UVG) dataset [33] in the same way as in Meta-SCI [45]. The former dataset is also the benchmark. but the compression ratio B is now set to 24. Because previous deep learning algorithms (including Meta-SCI) cannot scale for different compression ratios, traditional iteration algorithms including GAP-TV, PnP-FFDNet and PnP-FastDVDnet are chosen as baselines.

It can be noticed from Fig. 7 that these algorithms yeild worse results than ELP-unfolding meanwhile cost a longer time (details in SM). In the case of 1536×1536×12, PnP-FFDNet is able to get good results as ELP-unfolding. However, it is unstable and gets the worst results in the case of 256×256×24. Fig. 6 shows some selected images with much sharper boundaries and fewer

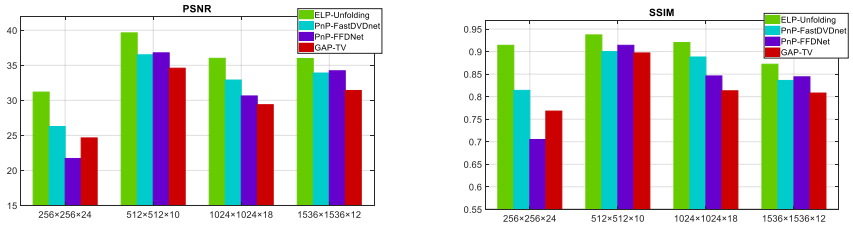

Fig. 7: Scalability: Reconstruction results by GAP-TV, PnP-FFDNet, PnP-FastDVDnet and the proposed ELP-unfolding with various spatial sizes and compression ratios.

artifacts reconstructed by ELP-unfolding than other algorithms. Please refer to the reconstructed videos in SM.

5.4 Ablation Study

In our ELP-unfolding model, the single prior period contains 8 stages, ensemble priors period contains 5 stages and thus the whole model contains 13 stages. Stage 9 has two priors to deal with projection operation while stage 10 has three priors and so on and so forth. In the end, in stage 13, there are six priors.

Focusing on the number of stages in Table 3b, we can see that the more stages one model has, the better result the model can achieve. But when the number of stages reaches 13, the reconstruction accuracy can not be improved any more. Regarding the priors, by adopting ensemble learning priors strategy, the 6 priors (with 13 stages) model can still improve reconstruction accuracy. Besides, the ensemble learning model always behaves better than its single prior counterpart in the same number of stage case. For instance, in the 9-stage model, two priors in the last stage always leads to better results than the single prior counterpart.

Next, we consider a more complicated structure. Specifically, we use 2 priors in stage 2 and 3 priors in stage 3 and so on and so forth. For a fair comparison, we also use a 13stage model. The result of this complicated model is called 'Ensemble all' in Table 3c. We can observe that even though the model is more complicated, it cannot lead to better results than our proposed structure, because the first several stages only provides rough estimates and the poor performance of first several priors can deteriorate the whole algorithm performance if coupled to the latter stages. In addition, the 'Ensemble no' in Table 3c denotes a single prior used in all stages, and the same for the 13 stages in Table 3b. This model can lead to decent results but not as good as ensemble priors structure. After comparison, we set the 6 priors model as our final ELP-unfolding, the results of which are also shown in 'Integrating all' in Table 3c.

Effect of dense connection. To verify the effect of the dense connection in ELP-unfolding, we make the comparisons with and without dense connection in the 7 stages model ('7-1') and 6 priors model ('13-6'). The 7 stages model has seven stages but each stage contains only one prior, while the 6-prior model is the full model in our paper that achieves SOTA results. As shown in Table 3d, removing the dense connection will lower the performance of unfolding

Table 3: Ablations. Average PSNR and SSIM for different setups in simulation.

(a) 'm-n' means m stages model has n priors in the last stage.

6 single stages		7 single stages		8 single stages	
'7-2' 32.69	'7-1' 30.71	'8-2' 32.82	'8-1' 30.98	'9-2' 33.24	'9-1' 31.25
'9-4' 32.92	'9-1' 31.25	'9-3' 33.03	'9-1' 31.25	'11-4' 33.33	'11-1' 31.45
'11-6' 33.06	'11-1' 31.45	'11-5' 33.19	'11-1' 31.45	'13-6' 33.46	'13-1' 31.75
'aver' 32.89	'aver' 31.14	'aver' 33.13	'aver' 31.23	'aver' 33.34	'aver' 31.48

(b) Different stages and ensemble priors in the last stage

1 stage	3 stages	5 stages	7 stages	8 stages	9 stages	11 stages	13 stages	1 prior	2 priors	4 priors	6 priors	8 priors
31.21, 0.926	33.29, 0.953	34.33, 0.964	34.50, 0.965	34.83, 0.966	34.92, 0.967	35.11, 0.968	35.07, 0.968	34.83, 0.966	34.98, 0.967	35.15, 0.968	35.41, 0.969	35.34, 0.969

(c) Running 13 stages in different situations.

Ensemble all	Ensemble no	Part training-set	Removing connection	Integrating all
34.97, 0.967	35.09, 0.968	34.73, 0.966	34.77, 0.966	35.41, 0.969

(d) '7-1' means 7 stages 1 prior while '13-6'means 13 stages 6 priors.

7-1 w/o connection	7-1 w/ connection	13-6 w/o connection	13-6 w/connection
34.23, 0.961	34.85, 0.967	34.77, 0.966	35.41, 0.969

algorithms including the single prior model and the ensemble prior model, because the information transmitted between priors is limited. It should be noticed that our dense connection operation is simple, only consisting of adding and concatenating, instead of complex operations such as the dense feature map adaption in Dense3D-Unfolding [46]. Thus our ELP-unfolding provides a simple strategy (dense connection) to improve the performance of deep unfolding.

Effect of ensemble priors. Table 3b can't completely reflect the effect of ensemble priors, because we adopt a large model by using connection technique and wide channels (512 channels in unet middle layer) to get SOTA accuracy to outperform Dense3D-Unfolding, which leaves little room for ensemble priors improvement. In most circumstances, it is unnecessary to use such big models. Thus, we use the normal 128 channels and remove connection technique to display the effect of ensemble priors, as shown in Table 3a. As we can see, the ensemble priors method can improve the reconstruction accuracy of PNSR by more than 1.75 dB on average. Besides, ensemble priors method doesn't increase memory to train and time to test. Thus, ensemble priors have a huge advantage in the field of deep unfolding.

Effect of training dataset. Training dataset plays a key role in performance of deep learning algorithms, and ELP-unfolding is no exception. We verify this by using part of the training dataset, *i.e.*the dataset in DAVIS2017 that only trains on 480p videos, but does not include the test dataset and test challenge dataset. The results are shown in 'Part training-set' in Table 3c. By comparing 'Part training-set' and 'Integrating all', we find that reducing the amount of training set hurts the performance of ELP-unfolding.

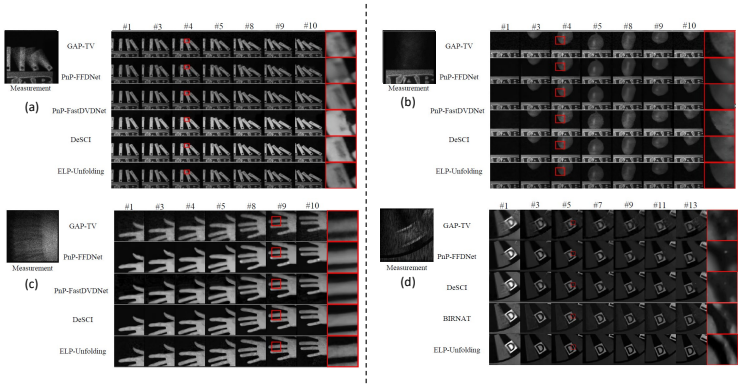

Fig. 8: Real data `duomino` (a, 512×512×10), `waterballon` (b, 512×512×10), `hand` (c, 512×512×10) and `chop` (d, 256×256×14) reconstructed from a compressed measurement.

6 Real datasets for SCI

We now apply the proposed ELP-unfolding to real datasets, namely chopwheel [30], waterBalloon [38], duomino [38] and hand [56]. Because of the unavoidable measurement noise, it is more challenging to reconstruct real measurements. The size of the Chopwheel data is 256×256×14, while the size of the other three datasets is 512×512×10. From Fig. 8, we can see that our method can generate more apparent contours while reducing artifacts and ghosting. What's more, previous deep learning algorithms didn't succeed in reconstructing hand because of the big noise in this data. Our ELP-unfolding firstly obtains the hand reconstruction by deep learning. Thus, we can only show the comparison with traditional iteration algorithms such as GAP-TV, PnP-FFDNet and DeSCI. Therefore, we can conclude that in practical applications, our method is powerful in reconstructing high-speed scenes. The relative videos can be seen in SM.

7 Conclusions and Future Work

Inspired by ensemble learning and iterative based optimization algorithm, we develop ensemble learning priors unfolding for scalable snapshot compressive imaging. Our ELP-unfolding algorithm has achieved state-of-the-art results in a short running time. Besides, we have firstly proposed the scalable function for SCI, not only in the spatial dimension but also in the temporal dimension.

To further improve the reconstruction accuracy, we will consider combining 3D-CNN with ELP-unfolding. Besides, to reduce the testing time and the parameters of neural network, a distilling method will be employed. We believe that our proposed ELP-unfolding framework can also be used for other inverse problems such as image CS, spectral compressive imaging, and so on [47, 53, 54, 62].

Acknowledgements: We would like to thank the Research Center for Industries of the Future (RCIF) at Westlake University, Westlake Foundation (2021B1501-2) and the funding from Lochn Optics.

References

1. Afonso, M.V., Bioucas-Dias, J.M., Figueiredo, M.A.: An augmented lagrangian approach to the constrained optimization formulation of imaging inverse problems. IEEE Transactions on Image Processing **20**(3), 681–695 (2010) 5
2. Akkus, Z., Galimzianova, A., Hoogi, A., Rubin, D.L., Erickson, B.J.: Deep learning for brain mri segmentation: state of the art and future directions. Journal of digital imaging **30**(4), 449–459 (2017) 3
3. Antipa, N., Oare, P., Bostan, E., Ng, R., Waller, L.: Video from stills: Lensless imaging with rolling shutter. In: 2019 IEEE International Conference on Computational Photography (ICCP). pp. 1–8. IEEE (2019) 3
4. Bioucas-Dias, J.M., Figueiredo, M.A.: A new twist: Two-step iterative shrinkage/thresholding algorithms for image restoration. IEEE Transactions on Image processing **16**(12), 2992–3004 (2007) 2, 3
5. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends in Machine Learning **3**(1), 1–122 (January 2011) 6
6. Cai, Y., Lin, J., Hu, X., Wang, H., Yuan, X., Zhang, Y., Timofte, R., Van Gool, L.: Mask-guided spectral-wise transformer for efficient hyperspectral image reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition i (2022) 3
7. Chen, Z., Zheng, S., Tong, Z., Yuan, X.: Physics-driven deep learning enables temporal compressive coherent diffraction imaging. Optica **9**(6), 677–680 (Jun 2022) 1
8. Cheng, Z., Chen, B., Liu, G., Zhang, H., Lu, R., Wang, Z., Yuan, X.: Memory-efficient network for large-scale video compressive sensing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16246–16255 (2021) 2, 7, 9, 10
9. Cheng, Z., Chen, B., Lu, R., Wang, Z., Zhang, H., Meng, Z., Yuan, X.: Recurrent neural networks for snapshot compressive imaging. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–1 (2022) 2
10. Cheng, Z., Lu, R., Wang, Z., Zhang, H., Chen, B., Meng, Z., Yuan, X.: Birnat: Bidirectional recurrent neural networks with adversarial training for video snapshot compressive imaging. In: European Conference on Computer Vision. pp. 258–275. Springer (2020) 2, 10
11. Dong, J., Fu, J., He, Z.: A deep learning reconstruction framework for x-ray computed tomography with incomplete data. PloS one **14**(11), e0224426 (2019) 3
12. Dong, W., Shi, G., Li, X., Ma, Y., Huang, F.: Compressive sensing via nonlocal low-rank regularization. IEEE transactions on image processing **23**(8), 3618–3632 (2014) 2, 3
13. Donoho, D.L.: Compressed sensing. IEEE Transactions on Information Theory **52**(4), 1289–1306 (April 2006) 5
14. Duarte, M.F., Davenport, M.A., Takhar, D., Laska, J.N., Sun, T., Kelly, K.F., Baraniuk, R.G.: Single-pixel imaging via compressive sampling. IEEE signal processing magazine **25**(2), 83–91 (2008) 5
15. Emmanuel, C., Romberg, J., Tao, T.: Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. IEEE Transactions on Information Theory **52**(2), 489–509 (February 2006) 5
16. Gibson, G.M., Sun, B., Edgar, M.P., Phillips, D.B., Hempler, N., Maker, G.T., Malcolm, G.P., Padgett, M.J.: Real-time imaging of methane gas leaks using a single-pixel camera. Optics express **25**(4), 2998–3005 (2017) 3

17. Gregor, K., LeCun, Y.: Learning fast approximations of sparse coding. In: Proceedings of the 27th international conference on machine learning. pp. 399–406 (2010) 2
18. He, W., Yokoya, N., Yuan, X.: Fast hyperspectral image recovery via non-iterative fusion of dual-camera compressive hyperspectral imaging. IEEE Transactions on Image Processing **30** (2021) 3
19. Higham, C.F., Murray-Smith, R., Padgett, M.J., Edgar, M.P.: Deep learning for real-time single-pixel video. Scientific reports **8**(1), 1–9 (2018) 3
20. Hitomi, Y., Gu, J., Gupta, M., Mitsunaga, T., Nayar, S.K.: Video from a single coded exposure photograph using a learned over-complete dictionary. In: 2011 International Conference on Computer Vision. pp. 287–294. IEEE (2011) 1
21. Hu, X., Cai, Y., Lin, J., Wang, H., Yuan, X., Zhang, Y., Timofte, R., Van Gool, L.: Hdnet: High-resolution dual-domain learning for spectral compressive imaging. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition i (2022) 3
22. Jalali, S., Yuan, X.: Snapshot compressed sensing: Performance bounds and algorithms. IEEE Transactions on Information Theory **65**(12), 8005–8024 (2019) 3, 5
23. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015) 9
24. Kulkarni, K., Lohit, S., Turaga, P., Kerviche, R., Ashok, A.: Reconnet: Non-iterative reconstruction of images from compressively sensed measurements. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 449–458 (2016) 2, 3
25. Li, C.: An efficient algorithm for total variation regularization with applications to the single pixel camera and compressive sensing. Rice University (2010) 5
26. Li, C., Yin, W., Jiang, H., Zhang, Y.: An efficient augmented lagrangian method with applications to total variation minimization. Computational Optimization and Applications **56**(3), 507–530 (2013) 2, 3
27. Lin, J., Cai, Y., Hu, X., Wang, H., Yuan, X., Zhang, Y., Timofte, R., Van Gool, L.: Coarse-to-fine sparse transformer for hyperspectral image reconstruction. arXiv preprint arXiv:2203.04845 (2022) 3
28. Liu, J., Pan, Y., Li, M., Chen, Z., Tang, L., Lu, C., Wang, J.: Applications of deep learning to mri images: A survey. Big Data Mining and Analytics **1**(1), 1–18 (2018) 3
29. Liu, Y., Yuan, X., Suo, J., Brady, D.J., Dai, Q.: Rank minimization for snapshot compressive imaging. IEEE transactions on pattern analysis and machine intelligence **41**(12), 2990–3006 (2018) 2, 3, 6, 9, 10
30. Llull, P., Liao, X., Yuan, X., Yang, J., Kittle, D., Carin, L., Sapiro, G., Brady, D.J.: Coded aperture compressive temporal imaging. Opt. Express **21**(9), 10526–10545 (May 2013) 1, 14
31. Meng, Z., Jalali, S., Yuan, X.: Gap-net for snapshot compressive imaging. arXiv preprint arXiv:2012.08364 (2020) 2, 3, 4, 10
32. Meng, Z., Ma, J., Yuan, X.: End-to-end low cost compressive spectral imaging with spatial-spectral self-attention. In: European Conference on Computer Vision. pp. 187–204. Springer (2020) 3
33. Mercat, A., Viitanen, M., Vanne, J.: Uvg dataset: 50/120fps 4k sequences for video codec analysis and development. In: Proceedings of the 11th ACM Multimedia Systems Conference. pp. 297–302 (2020) 11

34. Mousavi, A., Patel, A.B., Baraniuk, R.G.: A deep learning approach to structured signal recovery. In: 2015 53rd annual allerton conference on communication, control, and computing (Allerton). pp. 1336–1343. IEEE (2015) 2, 3
35. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. Advances in neural information processing systems **32**, 8026–8037 (2019) 9
36. Pintelas, P., Livieris, I.E.: Special issue on ensemble learning and applications. Algorithms **13**(6) (2020) 2
37. Pont-Tuset, J., Perazzi, F., Caelles, S., Arbeláez, P., Sorkine-Hornung, A., Van Gool, L.: The 2017 davis challenge on video object segmentation. arXiv preprint arXiv:1704.00675 (2017) 10
38. Qiao, M., Meng, Z., Ma, J., Yuan, X.: Deep learning for video compressive sensing. APL Photonics **5**(3), 030801 (2020) 1, 14
39. Qiao, M., Sun, Y., Ma, J., Meng, Z., Liu, X., Yuan, X.: Snapshot coherence tomographic imaging. IEEE Transactions on Computational Imaging **7**, 624–637 (2021) 3
40. Radwell, N., Johnson, S.D., Edgar, M.P., Higham, C.F., Murray-Smith, R., Padgett, M.J.: Deep learning optimized single-pixel lidar. Applied Physics Letters **115**(23), 231101 (2019) 3
41. Reddy, D., Veeraraghavan, A., Chellappa, R.: P2c2: Programmable pixel compressive camera for high speed imaging. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 329–336 (June 2011) 1
42. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: (MICCAI). LNCS, vol. 9351, pp. 234–241. Springer (2015) 7, 8
43. Shi, W., Jiang, F., Zhang, S., Zhao, D.: Deep networks for compressed image sensing. In: 2017 IEEE International Conference on Multimedia and Expo (ICME). pp. 877–882. IEEE (2017) 2, 3
44. Tassano, M., Delon, J., Veit, T.: Fastdvdnet: Towards real-time deep video denoising without flow estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1354–1363 (2020) 7
45. Wang, Z., Zhang, H., Cheng, Z., Chen, B., Yuan, X.: Metasci: Scalable and adaptive reconstruction for video compressive sensing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2083–2092 (2021) 2, 7, 8, 10, 11
46. Wu, Z., Zhang, J., Mou, C.: Dense deep unfolding network with 3d-cnn prior for snapshot compressive imaging. In: IEEE International Conference on Computer Vision (ICCV) (2021) 2, 3, 4, 7, 10, 13
47. Xue, Y., Zheng, S., Tahir, W., Wang, Z., Zhang, H., Meng, Z., Tian, L., Yuan, X.: Block modulating video compression: An ultra low complexity image compression encoder for resource limited platforms. CoRR (2022) 14
48. Yang, C., Qi, D., Cao, F., He, Y., Wang, X., Wen, W., Tian, J., Jia, T., Sun, Z., Zhang, S.: Improving the image reconstruction quality of compressed ultrafast photography via an augmented lagrangian algorithm. Journal of Optics **21**(3), 035703 (2019) 5
49. Yang, C., Yao, Y., Jin, C., Qi, D., Cao, F., He, Y., Yao, J., Ding, P., Gao, L., Jia, T., et al.: High-fidelity image reconstruction for compressed ultrafast photography via an augmented-lagrangian and deep-learning hybrid algorithm. Photonics Research **9**(2), B30–B37 (2021) 6

50. Yang, Y., Sun, J., Li, H., Xu, Z.: Deep admm-net for compressive sensing mri. In: Proceedings of the 30th international conference on neural information processing systems. pp. 10–18 (2016) 2, 3
51. Yang, Y., Sun, J., Li, H., Xu, Z.: Admm-csnet: A deep learning approach for image compressive sensing. IEEE transactions on pattern analysis and machine intelligence **42**(3), 521–538 (2018) 2, 3
52. Yoo, J., Sabir, S., Heo, D., Kim, K.H., Wahab, A., Choi, Y., Lee, S.I., Chae, E.Y., Kim, H.H., Bae, Y.M., et al.: Deep learning diffuse optical tomography. IEEE transactions on medical imaging **39**(4), 877–887 (2019) 3
53. Yuan, X.: Compressive dynamic range imaging via Bayesian shrinkage dictionary learning. Optical Engineering **55**(12), 123110 (2016) 14
54. Yuan, X., Liao, X., Llull, P., Brady, D., Carin, L.: Efficient patch-based approach for compressive depth imaging. Applied Optics **55**(27), 7556–7564 (Sep 2016) 14
55. Yuan, X.: Generalized alternating projection based total variation minimization for compressive sensing. In: 2016 IEEE International Conference on Image Processing (ICIP). pp. 2539–2543 (September 2016) 9, 10
56. Yuan, X., Brady, D.J., Katsaggelos, A.K.: Snapshot compressive imaging: Theory, algorithms, and applications. IEEE Signal Processing Magazine **38**(2), 65–88 (2021) 1, 14
57. Yuan, X., Jiang, H., Huang, G., Wilford, P.A.: Slope: Shrinkage of local overlapping patches estimator for lensless compressive imaging. IEEE Sensors Journal **16**(22), 8091–8102 (2016) 3
58. Yuan, X., Liu, Y., Suo, J., Dai, Q.: Plug-and-play algorithms for large-scale snapshot compressive imaging. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1447–1457 (June 2020) 1, 9, 10
59. Yuan, X., Liu, Y., Suo, J., Durand, F., Dai, Q.: Plug-and-play algorithms for video snapshot compressive imaging. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–1 (2021) 1, 10
60. Yuan, X., Llull, P., Liao, X., Yang, J., Brady, D.J., Sapiro, G., Carin, L.: Low-cost compressive sensing for color video and depth. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3318–3325 (2014) 1
61. Yuan, X., Pu, Y.: Parallel lensless compressive imaging via deep convolutional neural networks. Optics express **26**(2), 1962–1977 (2018) 3
62. Zhang, B., Yuan, X., Deng, C., Zhang, Z., Suo, J., Dai, Q.: End-to-end snapshot compressed super-resolution imaging with deep optics. Optica **9**(4), 451–454 (Apr 2022) 14
63. Zhang, J., Ghanem, B.: Ista-net: Interpretable optimization-inspired deep network for image compressive sensing. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1828–1837 (2018) 2, 4
64. Zhang, K., Zuo, W., Zhang, L.: Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. IEEE Transactions on Image Processing **27**(9), 4608–4622 (2018) 7
65. Zheng, H., Zhang, Y., Yang, L., Liang, P., Zhao, Z., Wang, C., Chen, D.Z.: A new ensemble learning framework for 3d biomedical image segmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 5909–5916 (2019) 2
66. Zheng, S., Liu, Y., Meng, Z., Qiao, M., Tong, Z., Yang, X., Han, S., Yuan, X.: Deep plug-and-play priors for spectral snapshot compressive imaging. Photon. Res. **9**(2), B18–B29 (Feb 2021) 3
67. Zheng, S., Wang, C., Yuan, X., Xin, H.L.: Super-compression of large electron microscopy time series by deep compressive sensing learning. Patterns **2**(7), 100292 (2021) 3

68. Zhou, K., Yang, Y., Qiao, Y., Xiang, T.: Domain adaptive ensemble learning. IEEE Transactions on Image Processing **30**, 8008–8018 (2021) 2