Ensemble Learning Priors Driven Deep Unfolding for Scalable Snapshot Compressive Imaging

Chengshuai Yang  
Westlake University  
integrityyang@gmail.com

Shiyu Zhang  
Westlake University  
zsylj@126.com

Xin Yuan  
Westlake University  
xyuan@westlake.edu.cn

Abstract

Snapshot compressive imaging (SCI) can record the 3D information by a 2D measurement and from this 2D measurement to reconstruct the original 3D information by reconstruction algorithm. As we can see, the reconstruction algorithm plays a vital role in SCI. Recently, deep learning algorithm show its outstanding ability, outperforming the traditional algorithm. Therefore, to improve deep learning algorithm reconstruction accuracy is an inevitable topic for SCI. Besides, deep learning algorithms are usually limited by scalability, and a well trained model in general can not be applied to new systems if lacking the new training process. To address these problems, we develop the ensemble learning priors to further improve the reconstruction accuracy and propose the scalable learning to empower deep learning the scalability just like the traditional algorithm. What’s more, our algorithm has achieved the state-of-the-art results, outperforming existing algorithms. Extensive results on both simulation and real datasets demonstrate the superiority of our proposed algorithm. The code and models will be released to the public. https://github.com/integritynoble/ELP-Unfolding/tree/master

1. Introduction

Recently, video snapshot compressive imaging (SCI) [22, 46] has attracted much attention because it can improve the imaging speed by capturing three-dimensional (3D) information from one 2D measurement. When video SCI works, multiple frames are first modulated by different masks, and these modulated frames are mapped into a single measurement. After this, the reconstruction algorithm recovers these multiple frames from the single measurement [43]. At present, the mask can easily be adjusted with a higher speed than the capture rate of the camera [14, 29, 31]. Thus, SCI enjoys the advantages of high speed, low memory, low bandwidth, low power and potentially low cost [44].

Figure 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, similar to traditional optimization-based algorithms, such as GAP-TV [42].

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 the traditional iterative reconstruction method not only in reconstruction accuracy but also in testing time [17, 25, 33, 39, 40]. But most deep learning methods lack interpretability. To increase interpretability, deep unfolding method has been developed, which simulates the traditional iteration algorithm [23, 35, 39, 40, 47]. Deep unfolding method adopts iterative framework but replaces traditional denoiser (such as total variation [3, 19] and nonlocal self-similarity [8, 21]) with the training neural network denoiser. So far, the deep unfolding method has achieved the best result for SCI. Among deep unfolding algorithms, GAP-net [23] can use the shortest time (0.0072 s) to achieve 32 dB for PSNR for benchmark dataset [21]. Dense3D-Unfolding [37] achieved best result (35 dB), though it costs a long time (1.35 s) because of using 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 the scalability. To apply the trained model to new systems, the model usually should be trained again. Although MetaSCI [36] can quickly be applied to new SCI modulations in spatial but not in temporal size, it still needs the adaptation (retraining) process.

Bearing the above concerns in mind, in order to achieve a higher reconstruction accuracy meanwhile with a high computing speed, we develop the Ensemble Learning Priors (ELP) unfolding based on 2D-CNN for SCI. Specifically, 2D-CNN can retain the fast processing and ensemble learning can increase reconstruction accuracy. Ensemble learning has a strong power to achieve the reconstruction accuracy and has also achieved the state-of-the-art (SOTA) results in a number of models for other tasks [27, 49, 50], due to the fact that several models have complementary advantages over a single model. Fortunately, the deep unfolding algorithm often contain many neural network priors, even though these priors stay in different iteration stages. Hereby, in this paper, we first propose to gather multiple neural network priors in one stage to realize ensemble learning for SCI without increasing training models. To further increase the reconstruction accuracy, the dense connection is added in the training models, which can help latter models learn some useful information from previous models. Thus our ELP-unfolding can achieve the SOTA result, better than Dense3D-Unfolding [37], and in a short running time.

Furthermore, to realize the scalability, we develop the scalable learning for SCI. Our method not only has scalability in the spatial dimensions but also in the temporal dimension. Considering spatial scalability, we set our ELP-unfolding without multilayer perception (MLP) structure. For dimensional scalability, the input of neural network priors are 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 the traditional iteration algorithms, to be applied to different systems. Our specific contributions 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 field, such as single pixel camera [12, 13, 30], MRI [2, 20] and tomography imaging [7, 41].

• We first propose the scalable learning for SCI, which can be used in the same way as the plug-and-play (PnP) algorithm [44]. For new systems with different modulations or different compression ratios, the trained model can still be applicable. Besides, scalable learning can achieve better results than PnP algorithm with a fast inference speed.

• We adopt the connection techniques in unfolding. In our ELP-unfolding, the connection technique just uses the adding and concatenating operation, which is simple. By contrast, the Dense3D-Unfolding [37] adopts a complex method such as DFMA (dense feature map adaption) to realize connection.

• Our method achieves the SOTA results for SCI in benchmark dataset based on 2D-CNN, outperforming the 3D-CNN method at a faster inference speed [37] as shown in Fig. 1.

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

2. Related Work

SCI is related to compressive sensing (CS) [15], where the reconstruction is significantly important as it provides the desired signals (such as images) from the compressed measurements. For CS [15, 17, 23, 25, 33], 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 [3, 19] denoiser has a fast speed but usually can only provide blurry images while the nonlocal self-similarity based denoiser [8, 21] can achieve a clearer image but takes a long time. Recently, deep learning has shown its strong power in reconstructing images [17, 25, 33, 39, 40]. 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 would contain the 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 been developed. Deep unfolding method simulates traditional iterative method and uses a lot of iterations (stages), each of which has projection operation and denoising operation. Different from traditional methods, deep unfolding used the trained neural network as a denoising prior. Therefore, deep learning mainly contributes to denoising in deep unfolding method and has little dependence on the 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 [23, 47]. Besides, Dense3D-Unfolding [37] obtained SOTA for SCI by combining deep unfolding method and 3D-CNN, but at the cost of a slow computing speed. Though the unfolding method can solve
the scalability problem in a variety of masks, the scalability problem in a variety of 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, we develop the ensemble learning priors unfolding for scalable SCI. We use the ensemble learning and 2D-CNN to realize the high reconstruction accuracy and speed and develop the scalable learning to realize scalability.

3. Preliminary: Video SCI System

As depicted in Fig. 2, let \( \{X_1, \ldots, X_B\} \) 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 \( \{C_1, \ldots, C_B\} \), respectively. The modulated frames are then integrated into a single coded measurement (a compressed image) \( Y \). Here, \( \{X_b\}_{b=1}^B \in \mathbb{R}^{n_x \times n_y \times B}, \{C_b\}_{b=1}^B \in \mathbb{R}^{n_x \times n_y \times B} \) and \( Y \in \mathbb{R}^{n_x \times n_y} \).

This forward model can be written as

\[
Y = \sum_{b=1}^B C_b \odot X_b + Z,
\]

(1)

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

\[
y = Hx + z,
\]

(2)

where \( y = \text{Vec}(Y) \in \mathbb{R}^{n_x n_y}, z = \text{Vec}(Z) \in \mathbb{R}^{n_x n_y} \) and \( x = \text{Vec}(X) = [\text{Vec}(X_1)^T, \ldots, \text{Vec}(X_B)^T]^T \in \mathbb{R}^{n_x n_y B} \).

Different from traditional compressive sensing [9–11], the sensing matrix \( H \) in (2) has a very special structure and can be written as

\[
H = [D_1, \ldots, D_B],
\]

(3)

where \( D_b = \text{diag}(\text{Vec}(C_b)) \in \mathbb{R}^{n_x n_y \times n_x n_y} \) 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 \( B > 1 \) [15].

4. Our proposed methods

4.1. Ensemble learning priors unfolding for SCI

Given the compressed measurement \( Y \) and coding pattern \( \{C_b\}_{b=1}^B \) captured by the SCI system, there exist two optimization frameworks to predict the desired high speed frames \( \{X_b\}_{b=1}^B \): penalty function method and augmented Lagrangian (AL) method. The performance of the AL method is better than that of the penalty function method, which has been proved in previous works [1, 18, 38], and therefore, the AL method is adopted here, which is formulated as follows:

\[
x = \arg\min_x \Phi(x) - \lambda_1 (y - Hx) + \frac{\gamma_1}{2} \|y - Hx\|_2^2,
\]

(4)

where \( \Phi(x), \lambda_1 \) and \( \gamma_1 \) denote the prior regularization, Lagrangian multiplier and penalty parameter, respectively. For convenience, Eq. (4) is further written as

\[
x = \arg\min_x \Phi(x) + \frac{\gamma_1}{2} \|y - Hx - \lambda_1 \|_2^2.
\]

(5)
4.1.1 Single prior

To solve Eq. (5), an auxiliary variable $v$ is introduced. Then Eq. (5) is further written as

$$x = \arg\min_x \Phi(v) + \frac{\lambda_1}{2} \left\| y - Hx - \frac{\lambda_1}{\gamma_1} \right\|_2^2$$

subject to $v = x$. \hfill (6)

By adopting alternating direction method of multipliers (ADMM) method \cite{4}, Eq. (6) is further written as

$$x, v = \arg\min_{x, v} \Phi(v) + \frac{\lambda_1}{2} \left\| y - Hx - \frac{\lambda_1}{\gamma_1} \right\|_2^2$$

$$+ \frac{\gamma_1}{2} \left\| v - x - \frac{\lambda_1}{\gamma_1} \right\|_2^2. \hfill (7)$$

According to ADMM, Eq. (7) can be divided into the two subproblems and solved iteratively

$$v^i = \arg\min_v \Phi(v) + \frac{\gamma_1}{2} \left\| v^{i-1} - v - \frac{\lambda_1}{\gamma_1} \right\|_2^2 \hfill (8)$$

$$x^i = \arg\min_x \frac{\lambda_1}{2} \left\| x - v^i - \frac{\lambda_1}{\gamma_1} \right\|_2^2 + \frac{\gamma_1}{2} \left\| y - Hx - \frac{\lambda_1}{\gamma_1} \right\|_2^2 \hfill (9)$$

where the superscript $i$ denotes the iteration index.

For subproblem $x_i$, there exists a closed-form solution, which is called projection operation

$$x^i = (\gamma_2^i + H^T \gamma_1^i H)^{-1} \left[ \lambda_1^i + \gamma_2^i v^i + H^T \gamma_1^i (y - \frac{\lambda_1^i}{\gamma_1^i}) \right]. \hfill (10)$$

Due to the special structure of $H$, this can be solved in one shot \cite{21}.

For subproblem $v_i$, Eq. (8) can be rewritten as

$$v^i = \arg\min_v \Phi(v) + \frac{\gamma_1}{2} \left\| v^{i-1} - v \right\|_2^2 \hfill (11)$$

where $u^{i-1} = x^{i-1} - \frac{\lambda_1^i}{\gamma_1^i}$. 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. 5.

4.1.2 Ensemble priors

In every stage of unfolding, the denoising prior has different parameters and thus plays different roles in removing noise, even if these priors have the same structure. To take full use of different denoisers among different stages, these priors after $m$ stages are gathered together to perform projection operation to produce $x$. Therefore, Eq. (8) and Eq. (9) in ensemble priors period becomes

$$v^{m+j} = \arg\min_v \Phi(v_i) + \frac{\gamma_1^{m+j}}{2} \left\| v^{m+j-1} - v - \frac{\lambda_1^{m+j-1}}{\gamma_1^{m+j-1}} \right\|_2^2 \hfill (12)$$

and

$$x^{m+j} = \arg\min_x \gamma_2^{m+j} \left\| x - v^{m+j} - \frac{\lambda_2^{m+j}}{\gamma_2^{m+j}} \right\|_2^2$$

$$+ \gamma_1^{m+j} \left\| y - Hx - \frac{\lambda_2^{m+j}}{\gamma_2^{m+j}} \right\|_2^2 + \cdots$$

$$+ \gamma_1^{m+j} \left\| y - Hx - \frac{\lambda_2^{m+j}}{\gamma_2^{m+j}} \right\|_2^2. \hfill (13)$$

For subproblem $v_i$, Eq. (12) can still adopt the same denoising prior form as in single-prior period. For subproblem $x_i$, there is slightly difference because of ensemble

$$x^{m+j} = (\gamma_2^m + \gamma_2^{m+1} + \cdots + \gamma_2^{m+j} + H^T \gamma_1^{m+j} H)^{-1} \left[ \lambda_2^m + \gamma_2^m v^m + \lambda_2^{m+1} + \gamma_2^{m+1} v^{m+1} + \cdots + \lambda_2^{m+j} + \gamma_2^{m+j} v^{m+j} + H^T \gamma_1^{m+j} (y - \frac{\lambda_2^{m+j}}{\gamma_2^{m+j}}) \right]. \hfill (14)$$

Last but not least, the Lagrangian multipliers $\lambda_1^i$ and $\lambda_2^i$ are updated by

$$\lambda_1^i = \lambda_1^{i-1} - \gamma_1^i (y - Hx^{i-1}), \hfill (15)$$

$$\lambda_2^i = \lambda_2^{i-1} - \gamma_2^i (x - v^{i-1}). \hfill (16)$$

Besides, the $\gamma_1^i$ and $\gamma_2^i$ are trained with the denoising prior parameters in every stage.

4.1.3 Denoising prior structure

As shown in Fig. 5, U-net \cite{32} is used as the backbone for denoising prior, which we adopt from FastDVDnet \cite{34}, but here we remove the 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 \cite{34, 48}, the penalty parameter $\gamma_2^i$ is expanded into a noise map as a part of the input. To help denoise, the normalized measurement $Y$ is also added into the input \cite{5, 36, 37}, which is defined as

$$Y = Y \odot \sum_{b=1}^{B} C_b. \hfill (17)$$
where $\odot$ represents the matrix element-wise division. Therefore, the input consists of noise map $\gamma_2$, normalized measurement $\mathbf{Y}$ and $x^{i-1} - \frac{\lambda_2^{i-1}}{\gamma_2}$, and the output is $v^i$. Besides, the dense connection is employed in the denoising prior network design.

![Figure 5. Denoising prior structure based on U-net [32].](image)

**Algorithm 1 ELP-unfolding for SCI Reconstruction**

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

1. Initialize $v^0 = 0$, $\lambda_1^0 = 0$, $\lambda_2^0 = 0$.
2. Update $x_0$ by Eq. (10).
3. % single prior period
   for $i = 1, \ldots, m$ do
   5. Update $v^i$ by Eq. (11). $\lambda_2^i$ by Eq. (16).
   6. Update $\lambda_1^i$ by Eq. (15), $x^i$ by Eq. (10).
   end for
4. % ensemble priors period
   for $k = m+1, \ldots, m+n$ do
   9. Update $v^k$ by Eq. (11). $\lambda_2^k$ by Eq. (16).
   10. Update $\lambda_1^k$ by Eq. (15), $x^k$ by Eq. (14).
   end for

4.1.4 Dense connection for unfolding

In traditional unfolding method, the connection between two stages are $\mathbf{v}$ and $\mathbf{u}$, that is $x^{i-1} - \frac{\lambda_2^{i-1}}{\gamma_2}$, which have a small number of temporal dimensions. Therefore, most latent information in U-net structure can not transmit between different priors. To break this bottleneck, the skip connection technique is used here. As shown in Fig. 5, in the $i^{th}$ prior, the feature $E_i^j$ and feature $E_{i-1}^{j+1}$ concatenate into one new feature, which will be as one whole feature operated in latent space in U-net structure. Besides, the feature $E_i^j$ will add to $E_{i+1}^{j+1}$ to form $E_{i+1}^{j+1}$, that is, $E_{i+1}^{j+1} = E_{i+1}^{j+1} + E_i^j$.

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

4.2. Scalable learning for SCI

The traditional deep learning methods usually do not have scalability. One trained model can only be applied to one system with specific masks and compression ratios $B$. When the scene data size changes, the new corresponding model usually needs to be trained again. The most recent MetaSCI [36] 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. Even some deep learning method, which is independent of multi-layer perception, such as Dense3D-Unfolding [5], can be applied to different spatial size case, but they have no temporal scalability. They must be trained again for new applications with different temporal dimensions i.e., $B$.

To address this problem, we develop the scalable learning for SCI. This scalable learning has scalability not only in the spatial dimension but also in temporal dimension. Specifically, to ensure spatial scalability, we only employ the convolutional neural network; to ensure temporal scalability, we first compute the maximum frames (temporal dimension) this model should have. For example, one model has $B$ frames. The input of the denoising prior should have $B + 2$ frames, as shown in Fig. 5. If the dynamic scene coincidentally has $B$ frames, the input can directly use original data. Otherwise, the original data should be rearranged several times to satisfy the frames number. When $B$ is not an integer multiple of the frame number of dynamic scene, only the first several frames of original data are used in the last arranging process. In the training process, the model randomly chooses different frames.

Even though the maximum temporal size needs to be set in front, 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 $\{C_b\}_{b=1}^B$, our ELP-unfolding can generate $\{\mathbf{X}_b\}_{b=1}^B \in \mathbb{R}^{n_y \times n_y \times B}$. The mean square error (MSE) is selected as our loss function, expressed as

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

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

We use PyTorch [26] to train our model on one NVIDIA A40 GPU. For all training processes, we adopt the same Adam optimizer [16] with a mini-batch size of 3 and a spatial size of $256 \times 256$. We also adopt the pre-training strategy. The whole training process has two periods. Firstly, 8 stages with a single prior model is trained to be used as the pretrained parameters. Secondly, the whole ELP-unfolding with 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. Besides, the former 8 stages in whole ELP-unfolding match the pretrained model very well and completely adopt the pretrained pa-
Table 1. Benchmark dataset: 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 the 6 benchmark datasets.

| Algorithm       | Kobe   | Traffic | Runner | Drop   | Crash  | Aerial | Average | Run time (s) |
|-----------------|--------|---------|--------|--------|--------|--------|---------|--------------|
| GAT-TV [12]     | 26.92  | 0.838   | 20.66  | 0.691  | 29.81  | 0.895  | 34.95   | 0.966        |
| DeSCI [21]      | 33.25  | 0.952   | 28.71  | 0.925  | 38.48  | 0.969  | 34.10   | 0.992        |
| PnP-FFDNet [44] | 30.33  | 0.925   | 24.01  | 0.835  | 32.44  | 0.931  | 39.68   | 0.986        |
| PnP-FastDVDNet [45] | 32.72  | 0.947   | 27.95  | 0.932  | 36.29  | 0.962  | 41.82   | 0.989        |
| BIRNAT [6]      | 32.71  | 0.950   | 29.33  | 0.942  | 38.70  | 0.976  | 27.84   | 0.932        |
| GAP-Unet-S12 [23] | 32.09  | 0.944   | 28.19  | 0.929  | 38.12  | 0.975  | 27.32   | 0.925        |
| Meta-SCI [36]   | 30.12  | 0.907   | 26.95  | 0.888  | 37.02  | 0.967  | 40.61   | 0.985        |
| RevSCI [5]      | 33.72  | 0.957   | 30.02  | 0.949  | 39.40  | 0.977  | 42.93   | 0.992        |
| Dense3D-Unfolding [37] | 35.00  | 0.969   | 31.76  | 0.966  | 30.46  | 0.980  | 44.96   | 0.995        |
| ELP-Unfolding (Ours) | 34.41  | 0.966   | 31.58  | 0.962  | 41.16  | 0.986  | 44.99   | 0.995        |

Video clips with spatial size of 256 × 256 are randomly cropped from this training dataset.

5.2. Benchmark datasets for SCI

The Kobe, Traffic, Runner, Drop, Crash, and Aerial are the Benchmark datasets for SCI [5, 6, 21, 37, 44, 45], where the data-size is 256 × 256 × 8, i.e., n x × n y × n z = 256, B = 8.

Based on these datasets, our ELP-unfolding with a special temporal size of 8 is been compared with other SOTA algorithms, including GAP-TV [42], DeSCI [21], PnP-FFDNet [44], PnP-FastDVDNet [45], BIRNAT [6], GAP-Unet-S12 [23], Meta-SCI [36], RevSCI [5], Dense3D-Unfolding [37]. 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 one second). Deep learning algorithms can achieve the better result in a short running time (usually less than 1 second). Although Dense3D-Unfolding had 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 the testing time (costing 0.24 s). For visualization purpose, we also present some images in Fig. 6, where we can see from the zooming areas that our ELP-unfolding provides much clearer images with sharper edges and more abundant details than other algorithms. We also believe that by adopting 3D-CCN, ELP-unfolding can achieve even better results.

5.3. Scalable datasets for SCI

To verify the scalability of our ELP-unfolding method, we 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 [24] in the same way as in Meta-SCI [36]. The former one dataset is also the benchmark but the compression ratio B is now set to 24. Because the previous deep learning algorithms (including Meta-SCI) cannot scale for different compression ratios, the traditional iteration algorithms including GAP-TV, PnP-FFDNet and PnP-FastDVDnet are chosen as the baselines. It can be no-
Figure 7. Scalability: Selected reconstruction results by GAP-TV, PnP-FFDNet, PnP-FastDVDNet and the proposed ELP-unfolding with various spatial sizes and compression ratios.

Figure 8. Scalability comparison based on PSNR and SSIM obtained by various algorithms for different spatial and temporal (compression ratio) sizes.

Figure 7 shows some selected images with much sharper boundaries and fewer artifacts reconstructed by ELP-unfolding than other algorithms. Please refer to the reconstructed videos in the SM.

5.4. Ablation Study

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

Focusing on the number of stages in Table 2, we can see that the more stages one model has, the better result this 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, in the same number of stage case, ensemble learning model always behaves better than its single prior counterpart. 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 13 stage model. The result of this complicated model is called ‘Ensemble all’ in Table 3. We can observe that even though the model is more complicated, it cannot lead to better results than our proposed structure. In addition, the ‘Ensemble no’ in Table 3 means a single prior used in all stages, and the same for the 13 stages in Table 2. This model can lead to decent results but not as good as our proposed structure. Thus, 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 3.

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 just contains one prior in every stage while 6 priors model is the full model in our paper to
Table 2. Ablation Study: Average PSNR and SSIM For different stages and ensemble priors (in the last stage) on the benchmark dataset.

| Stages | 1 stage | 3 stages | 5 stages | 7 stages | 9 stages | 11 stages | 13 stages | 1 prior | 2 priors | 4 priors | 6 priors | 8 priors |
|--------|---------|----------|----------|----------|----------|-----------|-----------|---------|----------|----------|----------|---------|
| PSNR   | 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 |

Table 3. Ablation Study: Average PSNR and SSIM for running 13 stages in different situations on benchmark datasets.

| Situation          | 13 stages |
|--------------------|-----------|
| Ensemble all       | 34.97, 0.967 |
| Ensemble no        | 35.09, 0.968 |
| Part training-set  | 34.73, 0.966 |
| Removing connection| 34.77, 0.966 |
| Integrating all    | 35.41, 0.969 |

Table 4. Ablation Study: Average PSNR and SSIM for connection between priors on the benchmark dataset. Here, ’7-1’ means seven stages one prior while ‘13-6’ means thirteen stages six priors.

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

achieve SOTA results. As shown in Table 4, removing the dense connection will lower the performance of unfolding 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, just consisting of adding and concatenating, instead of complex operations such as the dense feature map adaption in Dense3D-Unfolding [37]. Thus our ELP-unfolding provides a simple strategy (dense connection) to improve the performance of deep unfolding.

Effect of training dataset: The 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., only train dataset on 480p videos in DAVIS2017 but not including test dataset and test-challenge dataset; the results are shown in ‘Part training-set’ in Table 3. By comparing ‘Part training-set’ and ‘Integrating all’, we find reducing the amount of training set hurts the performance of ELP-unfolding.

5.5. Real datasets for SCI

We now apply the proposed ELP-unfolding to real datasets, namely, chopwheel [22], waterBalloon and duomino [29]. Because of the unavoidable measurement noise, it is more challenging to reconstruct real measurements. The chopwheel data is of size 256×256×14 while the other two datasets have size 512×512×10. From the Figs. 9, 10 and 11, we can see that our method can generate more apparent contours with fewer artifacts and ghosting. Therefore, we can conclude that in real applications, our method is powerful to reconstruct the desired high-speed scenes.

6. Conclusions and Future Work

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

To further improve the reconstruction accuracy, we will consider combining 3D-CNN into ELP-unfolding. Besides, to reduce the testing time and the parameters of neural network, the distilling method will be taken into use. We believe that our proposed ELP-unfolding framework can also be used in other inverse problems such as the image compressive sensing and hyperspectral compressive imaging.
References

[1] Manya V Afonso, José M Bioucas-Dias, and Mário AT Figueiredo. An augmented lagrangian approach to the constrained optimization formulation of imaging inverse problems. *IEEE Transactions on Image Processing*, 20(3):681–695, 2010. 3

[2] Zeynuttin Akkus, Alfiia Galimzianova, Assaf Hoogi, Daniel L Rubin, and Bradley J Erickson. Deep learning for brain mri segmentation: state of the art and future directions. *Journal of digital imaging*, 30(4):449–459, 2017. 2

[3] José M Bioucas-Dias and Mário AT Figueiredo. A new twist: Two-step iterative shrinkage/thresholding algorithms for image restoration. *IEEE Transactions on Image processing*, 16(12):2992–3004, 2007. 1, 2

[4] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, January 2011. 4

[5] Ziheng Cheng, Bo Chen, Guanliang Liu, Hao Zhang, Ruiying Lu, Zhengjue Wang, and Xin Yuan. Memory-efficient network for large-scale video compressive sensing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16246–16255, 2021. 4, 5, 6

[6] Ziheng Cheng, Ruiying Lu, Zhengjue Wang, Hao Zhang, Bo Chen, Ziyi Meng, and Xin Yuan. Birnart: Bidirectional recurrent neural networks with adversarial training for video snapshot compressive imaging. In *European Conference on Computer Vision*, pages 258–275. Springer, 2020. 6

[7] Jianbing Dong, Jian Fu, and Zhao He. A deep learning reconstruction framework for x-ray computed tomography with incomplete data. *PloS one*, 14(11):e0224426, 2019. 2

[8] Weisheng Dong, Guangming Shi, Xin Li, Yi Ma, and Feng Huang. Compressive sensing via nonlocal low-rank regularization. *IEEE transactions on image processing*, 23(8):3618–3632, 2014. 1, 2

[9] D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, April 2006. 3

[10] Marco F Duarte, Mark A Davenport, Dharmalak Takhair, Jason N Laska, Ting Sun, Kevin F Kelly, and Richard G Baraniuk. Single-pixel imaging via compressive sampling. *IEEE signal processing magazine*, 25(2):83–91, 2008. 3

[11] Candes Emmanuel, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE transactions on Information Theory*, 52(2):489–509, February 2006. 3

[12] Graham M Gibson, Baoqin Sun, Matthew P Edgar, David B Phillips, Nils Hempler, Gareth T Maker, Graeme PA Malcolm, and Miles J Padgett. Real-time imaging of methane gas leaks using a single-pixel camera. *Optics express*, 25(4):2998–3005, 2017. 2

[13] Catherine F Higham, Roderick Murray-Smith, Miles J Padgett, and Matthew P Edgar. Deep learning for real-time single-pixel video. *Scientific reports*, 8(1):1–9, 2018. 2

[14] 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

[15] Shirin Jalali and Xin Yuan. Snapshot compressed sensing: Performance bounds and algorithms. *IEEE Transactions on Information Theory*, 65(12):8005–8024, 2019. 2, 3

[16] Diederek P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[17] Kuldeep Kulkarni, Suhas Lohit, Pavan Turaga, Ronan Kerlvihe, 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. 1, 2

[18] Chengbo Li. An efficient algorithm for total variation regularization with applications to the single pixel camera and compressive sensing. Rice University, 2010. 3

[19] Chengbo Li, Wotao Yin, Hong Jiang, and Yin Zhang. An efficient augmented lagrangian method with applications to total variation minimization. *Computational Optimization and Applications*, 53(3):507–530, 2013. 1, 2

[20] Jin Liu, Yi Pan, Min Li, Ziyue Chen, Lu Tang, Chengqian Lu, and Jianxin Wang. Applications of deep learning to mri images: A survey. *Big Data Mining and Analytics*, 1(1):1–18, 2018. 2

[21] Yang Liu, Xin Yuan, Jini 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, 4, 6

[22] Patrick Llull, Xuejun Liao, Xin Yuan, Jianbo Yang, David Kittle, Lawrence Carin, Guillermo Sapiro, and David J. Brady. Coded aperture compressive temporal imaging. *Opt. Express*, 21(9):10526–10545, May 2013. 1, 8

[23] Ziyi Meng, Shirin Jalali, and Xin Yuan. Gap-net for snapshot compressive imaging. *arXiv preprint arXiv:2012.08364*, 2020. 1, 2, 6

[24] Alexandre Mercat, Marko Viitanen, and Jarno Vanne. Uvg dataset: 50/120fps 4k sequences for video codec analysis and development. In *Proceedings of the 11th ACM Multimedia Systems Conference*, pages 297–302, 2020. 6

[25] Ali Mousavi, Ankit B Patel, and Richard G Baraniuk. A deep learning approach to structured signal recovery. In *2015 53rd annual allerton conference on communication, control, and computing (Allerton)*, pages 1336–1343. IEEE, 2015. 1, 2

[26] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037, 2019. 5

[27] Panagiotis Pintelas and Ioannis E Livieris. Special issue on ensemble learning and applications, 2020. 2

[28] Jordi Pont-Tuset, Federico Perazzi, Sergio Caelles, Pablo Arbelàez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017. 6
[29] Mu Qiao, Ziyi Meng, Jiawei Ma, and Xin Yuan. Deep learning for video compressive sensing. *APL Photonics*, 5(3):030801, 2020. 1, 8

[30] Neal Radwell, Steven D Johnson, Matthew P Edgar, Catherine F Higham, Roderick Murray-Smith, and Miles J Padgett. Deep learning optimized single-pixel lidar. *Applied Physics Letters*, 115(23):231101, 2019. 2

[31] Dikpal Reddy, Ashok Veeraraghavan, and Rama Chellappa. P2c2: Programmable pixel compressive camera for high speed imaging. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 329–336, June 2011. 1

[32] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, volume 9351 of LNCS, pages 234–241. Springer, 2015. 4, 5

[33] Wuzhen Shi, Feng Jiang, Shengping Zhang, and Debin Zhao. Deep networks for compressed image sensing. In *2017 IEEE International Conference on Multimedia and Expo (ICME)*, pages 877–882. IEEE, 2017. 1, 2

[34] Matias Tassano, Julie Delon, and Thomas Veit. Fastdvdnet: Towards real-time deep video denoising without flow estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1354–1363, 2020. 4

[35] Junxiang Wang, Fuxun Yu, Xiang Chen, and Liang Zhao. Admm for efficient deep learning with global convergence. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 111–119, 2019. 1

[36] Zhengjue Wang, Hao Zheng, Yizhe Zhang, Lin Yang, Peixian Liang, Zhuo Sun, and Shian Zhang. Improving the image reconstruction quality of compressed ultrafast photography via an augmented lagrangian algorithm. *Journal of Optics*, 21(3):035703, 2019. 3

[37] Chengshuai Yang, Dalong Qi, Fengyan Cao, Yilin He, Xing Wang, Wenlong Wen, Jinshou Tian, Tianqing Jia, Zhenrong Sun, and Shian Zhang. Improving the image reconstruction quality of compressed ultrafast photography via an augmented lagrangian algorithm. *Journal of Optics*, 21(3):035703, 2019. 3

[38] Yan Yang, Jian Sun, Huibin Li, and Zongben Xu. Deep admm-net for compressive sensing mri. In *Proceedings of the 30th international conference on neural information processing systems*, pages 10–18, 2016. 1, 2

[39] Yan Yang, Jian Sun, Huibin Li, and Zongben Xu. Admmcsnet: A deep learning approach for image compressive sensing. *IEEE transactions on pattern analysis and machine intelligence*, 42(3):521–538, 2018. 1, 2

[40] Jiaeun Yoo, Sohail Sabir, Duchang Heo, Kee Hyun Kim, Abdul Wahab, Yoonsok Choi, Seul-I Lee, Eun Young Chae, Hak Hee Kim, Young Min Bae, et al. Deep learning diffuse optical tomography. *IEEE transactions on medical imaging*, 39(4):877–887, 2019. 2

[41] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *arXiv preprint arXiv:2109.06548*, 2021. 1, 2, 4, 6

[42] Xin Yuan. Generalized alternating projection based total variation minimization for compressive sensing. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 2539–2543, September 2016. 1, 6