A NEW BACKBONE FOR HYPERSPECTRAL IMAGE RECONSTRUCTION

Anonymous authors
Paper under double-blind review

ABSTRACT

The study of 3D hyperspectral image (HSI) reconstruction refers to the inverse process of snapshot compressive imaging, during which the optical system, *e.g.*, the coded aperture snapshot spectral imaging (CASSI) system, captures the 3D spatial-spectral signal and encodes it to a 2D measurement. While numerous sophisticated neural networks have been elaborated for end-to-end reconstruction, trade-offs still need to be made among **performance**, **efficiency** (training and inference time), and **feasibility** (the ability of restoring high resolution HSI on limited GPU memory). This raises a challenge to design a new baseline to conjointly meet the above requirements. In this paper, we fill in this blank by proposing a Spatial/Spectral Invariant Residual U-Net, namely SSI-ResU-Net. It differentiates with U-Net in three folds–1) scale/spectral-invariant learning, 2) nested residual learning, and 3) computational efficiency. Benefiting from these three modules, the proposed SSI-ResU-Net outperforms the current state-of-the-art method TSA-Net Meng et al. (2020b) by over 3 dB in PSNR and 0.036 in SSIM while only using 2.82% trainable parameters. To the greatest extent, SSI-ResU-Net achieves competing performance with over 77.3% reduction in terms of floating-point operations (FLOPs), which for the first time, makes high-resolution HSI reconstruction feasible under practical application scenarios.

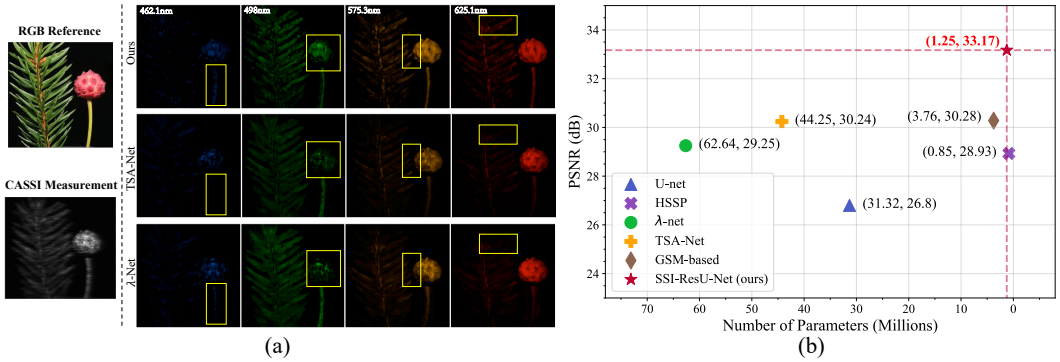
1 INTRODUCTION

Hyperspectral imaging refers to multi-channel imaging where each channel stores information for a fixed real-world scene at a specific spectral wavelength Plaza et al. (2009). By capturing spatial intensity in a spectral-wise manner, hyperspectral images empower richer information than normal RGB image cubes and they have been applied in a wide range of scenarios, *e.g.*, object detection Kim et al. (2012); Xu et al. (2015), remote sensing Borengasser et al. (2007); Melgani & Bruzzone (2004); Yuan et al. (2017), medical image processing Lu & Fei (2014); Meng et al. (2020c) etc. HSI can be captured and measured by snapshot compressive imaging (SCI) systems, tending to compress information of snapshots along the spectral axis into one single 2D measurement Yuan et al. (2021). The coded aperture snapshot spectral imaging (CASSI) system Wagadarikar et al. (2008); Meng et al. (2020b) forms one mainstream research direction among existing SCI systems Llull et al. (2013); Wagadarikar et al. (2008; 2009); Yuan et al. (2015).

Based on the 2D measurements generated by CASSI, a quite number of reconstruction algorithms Bioucas-Dias & Figueiredo (2007); Liu et al. (2019); Miao et al. (2019); Meng et al. (2020b); Wang et al. (2020; 2017; 2019); Yuan (2016a) have been proposed to recover the original hyperspectral signal. The goal of HSI reconstruction is to transform the given input 3D cubes (denoted as $\boldsymbol{F}$ in the top-left region of Fig. 2) into the desired hyperspectral images, resulting in a mapping function underlying high-dimension data space. Deep neural networks enable an effective way to approximate such an end-to-end mapping relationship. For example, some deep HSI reconstruction networks Meng et al. (2020b); Miao et al. (2019); Wang et al. (2019) have been proposed recently, which have achieved impressive performance on both synthetic and real HSI data.

1.1 MOTIVATION

Despite the success of recent neural networks-based HSI reconstruction methods, several deep-seated challenges cast a shadow on future development.



(a) (b)

Figure 1: (a) Reconstructed HSI using our model (first row on the right), compared with previous state-of-the-art TSA-Net Meng et al. (2020b) (middle-right) and λ-net Miao et al. (2019) (bottom-right) from the CASSI measurement (bottom-left). The RGB image (top-left) is shown as a reference to demonstrate the color. The reconstructed details are highlighted with yellow bounding boxes (zoom in for clear comparison). (b) Comparison between different HSI reconstruction methods in terms of PSNR (y-axis) and model size (x-axis). The x-axis is reversed for better visualization.

1) A large HSI dataset including considerable training samples and abundant spectral wavelength is currently inaccessible. This inevitably limits the learning capacity of some powerful network architectures, *e.g.*, the transformers in Meng et al. (2020b), self-attention generative adversarial network in Ma et al. (2019). To substantially avoid overfitting, reducing complexity of reconstructive network is a sound direction.
2) Among existing end-to-end baselines, U-Net Ronneberger et al. (2015) has been widely used as a backbone by recent researchers. However, it is initially proposed for biomedical image processing, which differentiates from HSI reconstruction in many perspectives.
3) Compared with U-Net, existing deep models achieve better performance but with more extensive parameters, leading to tremendous training time and inference time. For example, λ-net Miao et al. (2019) and TSA-Net Meng et al. (2020b) need weeks of training time.
4) Given a neural network structure, higher-resolution inputs mean increasing floating-point operations (FLOPs). Considering limited computational resources (i.e., GPU memory) under real conditions, it's hard to flexibly train or deploy exsiting DCNN models for inputs with large spatial size (i.e., TSA-Net takes up over 18GB GPU memory for $1024\times1024\times28$ HSI restoration).

1.2 Contributions of This Work

To address the above challenges, we focus on discussing the potential of using a *concise* network to handle the HSI reconstruction problem from several insights. We propose a simple yet surprisingly efficient CNN reconstructive network, called SSI-ResU-Net (see Fig. 2), by building on top of a the U-Net structure. The contributions of this work are summarized as follows.

- In response to the demand of HSI-specialized reconstructive backbone, we further propose a method, SSI-ResU-Net, based on the U-Net Ronneberger et al. (2015). Although our modulation strategies (*i.e.*, nested residual learning, spatial/spectral invariance, computation minimization.) have been leveraged in other topics, they haven't been well explored and developed for the HSI reconstruction problem. This study empirically shows that the proposed SSI-ResU-Net significantly improves HSI reconstruction by reutilizing these well-known techniques.
- The proposed SSI-ResU-Net sets a state-of-the-art by improving the most recent method Meng et al. (2020b) over 3 dB in PSNR. Extensive experimental results on simulation and real data captured by the CASSI system have been provided to validate the performance. Moreover, our approach presents clear perceptual improvement across different spectral channels (*e.g.*, Fig. 1 (a)).
- Without introducing complicated modules, our approach significantly shrinks the model size (see Fig. 1 (b)) without sacrifice the performance, *i.e.*, to most extent, we only using 2.82% parameters of TSA-Net Meng et al. (2020b). Therefore, the efficiency has been

greatly improved, *i.e.*, the training time of SSI-ResU-Net only takes less than two days on a TIAN RTX GPU.

- Based on the proposed small-size DCNN model, to reduce the amount of computation, we removed the Batch Normalization layers and further employed the re-scaling operations. In total, we reduce over 77.3% FLOPs reduction (see Table 2) and much lower GPU usage. For the first time, this makes higher-resolution images reconstruction possible under normal GPU conditions.

2 Related Work

The basic idea of SCI is to modulate the high-dimensional signal using a higher frequency than capture rate of the camera. In this manner, a compressed coded frame captured by the camera will include the information in the high-dimensional signal and a high-performance algorithm can then be employed to recover the desired data. As a novel implementation of SCI, CASSI uses a coded aperture and a prism to implement the spectral modulation.

Inspired by the success of deep learning in other image translation problems, researchers have started using deep learning to reconstruct hyperspectral images from CASSI measurements Meng et al. (2020b;c); Miao et al. (2019); Wang et al. (2019); Wang et al. (2019; 2020); Zheng et al. (2021), which can be substantially divided into two streams: E2E-NN models and others. The former stream tends to directly learn a complete mapping function from measurements (always packaged with masks) to estimated HSIs. Other researchers turns to introduce CNN models into conventional optimization algorithms, named unrolling/unfolding methods, leading to lightweight and interpretable methods. Coincidentally, U-Net Ronneberger et al. (2015) has been deemed as a reconstructive backbone and widely used in both E2E and non-E2E models. For example, the λ-net Miao et al. (2019) is a dual-stage generative model which employs a U-Net and residual learning strategy. The TSA-Net Meng et al. (2020b) which combined spatial-spectral self-attention with U-Net led to excellent results on both simulation and real data. As a deep unfolding method, GAP-net Meng et al. (2020a) utilizes 15-layer U-Net as a trained denoiser in each stage. Recent Gaussian Scale Mixture Prior-based (GSM-based) baseline Huang et al. (2021) employing U-Net for two objects: a lightweight U-Net for approximating the regularization parameters, another lightweight U-Net for estimating the local-mean of GSM prior.

Since proposed in Ronneberger et al. (2015), U-Net has become a classical model structure for biomedical image segmentation, followed by which many variants have been proposed in recent years, *i.e.*, 3D U-Net Çiçek et al. (2016), Ternaus U-Net Iglovikov & Shvets (2018), MultiResUNet Ibtehaz & Rahman (2020) and Attention U-Net Oktay et al. (2018) etc. The reason that U-Net show extraordinary talents in this field is that it highly adapted to characteristic of medical images: The semantic information contained in medical images is generally regular and traceable, which can be easily abstracted in coarse (low-resolution) feature map. The medical images contains blurry boundaries and complex gradient distribution, which can only been captured via high-resolution feature map. U-Net combines low-resolution feature map and high-resolution feature map via multiple concatenations and thus perfectly matched with medical images.

However, for the purpose of HSI restoration, U-Net only achieves sub-optimal performance when being solely referred to as a reconstructive baseline, this indicates that it may not be a prime candidate for subsequent exploitation. Researchers thus generally package U-Net into larger models and make efforts out of U-net, but devotes no attention inside the U-Net, *i.e.*, directly changes U-Net, according to insightful analysis. It turns out that performance of neural networks are sensitive to minor adjustments, there exists a "varient" of U-Net that enables a significant performance boost is a reasonable assumption. In this paper, firstly, to acquire better reconstructive performance, we carefully modify the existing U-Net architecture by substituting improper partitions with reconstruction-oriented components. Secondly, to ultimately improve the efficiency and the feasibility of the proposed model, we further cut off inefficient modules and take actions to reduce FLOPs in the model.

3 Methodology

To achieve our goal of reconstruction, firstly the hyperspectral signal needs to be captured by optical apparatus in CASSI system. As demonstrated in Fig. 2 top-left region, three-dimensional signal

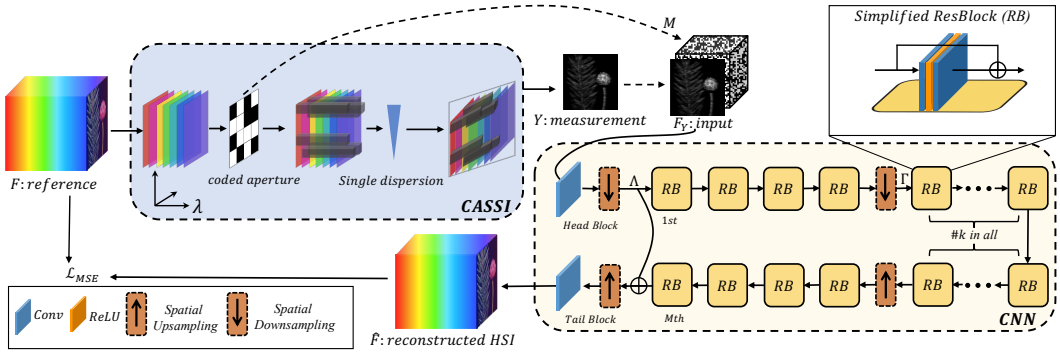


Figure 2: Flow-chart of compression (CASSI) and corresponding inverse process (DCNN). In top-left blue box, a 2D coded aperture and a disperser of CASSI system is drawn here while other optical components are omitted. For synthetic HSI cube $\boldsymbol{F}$, measurement $\boldsymbol{Y}$ can be mathematically computed by modeling CASSI system. For real-world unknown HSI signal $\boldsymbol{F}$, measurement $\boldsymbol{Y}$ is captured by a real CASSI system proposed in Meng et al. (2020b). In bottom-right yellow box, our proposed reconstructive model structure is outlined. Masks $\boldsymbol{M}$ (shifted variants of the same one) and the measurement are merged and further sent into proposed model f_{res}, which is composed of head block f_{head}, main body f_M and tail block f_{tail}. The main body learns the residual mapping between the input and output. The embedding Λ is of 128×128 and embedding Γ is of 64×64.

corresponding to a given real-world (synthetic) scene is compressed into a 2D measurement. Then we conduct dot productions between the measurement and each spectral channels of shifted variants of coded aperture, namely mask. For reconstruction, as HSI shares little similar characteristics with biomedical images, there's no need to keep contracting-expansive structure and multiple concatenations. Inspired by previous works in HSI pixel-wise classification Roy et al. (2020); Tao et al. (2015), we regard the NN model as a straightforward feature transformer that jointly processes spectral-spatial information. For simplicity, we propose a spatial/spectral scale invariant fashion. Nested residual learning is introduced to empower the learning capacity. The learning objective of the reconstructive model is to minimize the difference between reference and CNN outputs.

3.1 Mathematical Model of CASSI

Given the 3D spectral cube (Fig. 2 top-left), denoted by $\boldsymbol{F} \in \mathbb{R}^{N_x \times N_y \times N_\lambda}$, where N_x, N_y, and N_λ represent the spectral image's height, width, and total number of wavelengths, respectively, we compute signal modulation (implemented by the mask) in a channel-wisely way as

$$\boldsymbol{F}'(:,:,n_\lambda) = \boldsymbol{F}(:,:,n_\lambda) \odot \boldsymbol{M}^*, \tag{1}$$

where $\boldsymbol{F}' \in \mathbb{R}^{N_x \times N_y \times N_\lambda}$ represents the modulated signals, $\boldsymbol{M}^* \in \mathbb{R}^{N_x \times N_y}$ refers to a pre-defined physical mask governing on signal modulation, $n_\lambda \in [1, \ldots, N_\lambda]$ indexes wavelengths, and $\odot$ is the Hadamard product.

By passing $\boldsymbol{F}'$ to a disperser, the cube $\boldsymbol{F}'$ becomes tilted and could be sheared along the y-axis. Let $\boldsymbol{F}'' \in \mathbb{R}^{N_x \times (Ny + N_\lambda - 1) \times N_\lambda}$ be the tilted cube, and λ_c be a reference wavelength, *i.e.*, $\boldsymbol{F}'[:,:,n_{\lambda_c}]$ works like an anchor image that is not sheared along the y-axis, we have

$$\boldsymbol{F}''(u, v, n_\lambda) = \boldsymbol{F}'(x, y + d(\lambda_n - \lambda_c), n_\lambda), \tag{2}$$

where (u, v) locates the coordinate system on the detector plane, λ_n denotes the n_λ-th channel, λ_c refers to the anchored wavelength, and $d(\lambda_n - \lambda_c)$ represents a spatial shift of the n_λ-th channel in $\boldsymbol{F}'$. Notably, since the sensor integrates all the light within the wavelength range $[\lambda_{\min}, \lambda_{\max}]$, we could model the compressed measurement at a detector $y(u, v)$ with equation 2 and the following integral:

$$y(u, v) = \int_{\lambda_{\min}}^{\lambda_{\max}} f''(u, v, n_\lambda) \mathrm{d}\lambda, \tag{3}$$

where f'' is the analog (continuous) representation of $\boldsymbol{F}''$. To discretize equation 3, we further have

$$\boldsymbol{Y} = \sum_{n_\lambda=1}^{N_\lambda} \boldsymbol{F}''(:,:,n_\lambda) + \boldsymbol{G}. \tag{4}$$

$\boldsymbol{Y} \in \mathbb{R}^{N_x \times (Ny+N_\lambda-1)}$ represents a 2D measurement, which, in essence, is a *compressed* frame capturing the information, and $\boldsymbol{G} \in \mathbb{R}^{N_x \times (Ny+N_\lambda-1)}$ is a corresponding 2D measurement noise.

To simplify model notations, we define $\boldsymbol{M} \in \mathbb{R}^{N_x \times (Ny+N_\lambda-1) \times N_\lambda}$ and $\tilde{\boldsymbol{F}} \in \mathbb{R}^{N_x \times (Ny+N_\lambda-1) \times N_\lambda}$ as the shifted masks and signal frames of different wavelengths as the following:

$$\begin{aligned} \boldsymbol{M}(u, v, n_\lambda) &= \boldsymbol{M}^*(x, y + d(\lambda_n - \lambda_c)), \\ \tilde{\boldsymbol{F}}(u, v, n_\lambda) &= \boldsymbol{F}(x, y + d(\lambda_n - \lambda_c), n_\lambda). \end{aligned} \tag{5}$$

By using $\boldsymbol{M}$ and $\tilde{\boldsymbol{F}}$, the measurement $\boldsymbol{Y}$ in equation 4 could be reformulated as

$$\boldsymbol{Y} = \textstyle\sum_{n_\lambda=1}^{N_\lambda} \tilde{\boldsymbol{F}}(:, :, n_\lambda) \odot \boldsymbol{M}(:, :, n_\lambda) + \boldsymbol{G}. \tag{6}$$

Vectorization. Let $\boldsymbol{y} = \text{vec}(\boldsymbol{Y})$ and $\boldsymbol{g} = \text{vec}(\boldsymbol{G}) \in \mathbb{R}^n$ be the vectorization of matrices $\boldsymbol{Y}$ and $\boldsymbol{G}$, where $\text{vec}(\cdot)$ concatenates all the columns of a matrix as one single vector. Similarly, we have $\tilde{\boldsymbol{f}}^{(n_\lambda)} = \text{vec}(\tilde{\boldsymbol{F}}(:, :, n_\lambda))$, resulting in the vector $\boldsymbol{f} = \text{vec}([\tilde{\boldsymbol{f}}^{(1)} \dots \tilde{\boldsymbol{f}}^{(N_\lambda)}]) \in \mathbb{R}^{nN_\lambda}$, where $n = N_x(N_y + N_\lambda - 1)$. By defining the sensing matrix as

$$\boldsymbol{\Phi} = [\boldsymbol{D}_1, \dots, \boldsymbol{D}_{N_\lambda}] \in \mathbb{R}^{n \times nN_\lambda}, \tag{7}$$

where $\boldsymbol{D}_{n_\lambda} = \text{Diag}(\text{vec}(\boldsymbol{M}(:, :, n_\lambda)))$ is a diagonal matrix expanded by $\text{vec}(\boldsymbol{M}(:, :, n_\lambda))$, rewrite equation 6 as

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{f} + \boldsymbol{g}. \tag{8}$$

While obtaining the vectorized measure $\boldsymbol{y}$ with equation 8 is similar to compressive sensing Donoho (2006); Emmanuel et al. (2006), the sensing matrix $\boldsymbol{\Phi}$ given in equation 7 has a very special structure that most existing well-developed compressive sensing theories cannot easily handle, *i.e.*, $\boldsymbol{\Phi}$ is not only a fat matrix but also highly sparse—at most nN_λ nonzero elements. However, it is worth noting that, the recent studies Jalali & Yuan (2018); Jalali & Yuan (2019) have shown the signal can still be recovered even when $N_\lambda > 1$.

Given the measurement $\boldsymbol{y}$ captured by the camera and $\boldsymbol{\Phi}$ calibrated upon pre-design, one critical and practical problem of CASSI is to solve $\boldsymbol{f}$ used in equation 8, falling in the vein of inverse problem Yuan et al. (2021), for which we will propose a simple yet surprisingly efficient deep neural network below.

3.2 SSI-ResU-Net: Beyond the U-Net Architecture

We use $\boldsymbol{F}_Y \in \mathbb{R}^{N_x \times Ny \times N_\lambda}$ as the initialized input for our proposed network, and define its n_λ-th channel as

$$\boldsymbol{F}_Y[:, :, n_\lambda] := shift(\boldsymbol{M}_{n_\lambda} \odot \boldsymbol{Y}), \tag{9}$$

where $\odot$ is a Hadamard product, and $shift$ means raw output will be shifted into the desired tensor shape. Overall, the proposed model aims to learn the mapping function as

$$f_{res}(\cdot) : \boldsymbol{F}_Y \to \widehat{\boldsymbol{F}}, \tag{10}$$

where $\widehat{\boldsymbol{F}}$ denotes the reconstruction result. Essentially, our proposed network $f_{res}(\cdot)$ learns how to fit $\boldsymbol{\Phi}^{-1}$ expressed in a vector space, given by equation 8, during the simulation. We suppose that it's possible to train a tiny but promising E2E model to learn the mapping function, given paired data. Considering the simple U-net architecture has already achieved sub-optimal performance (as compared in Tab. 1), building model structure upon it is a potential choice. The rest is to carefully do modification. Note that our proposed model structure, SSI-ResU-Net, contains much difference with U-Net regarding detailed model structures. As it origins from U-Net and maintains a symmetrical structure, we keep an "U" in the name.

As shown in Fig. 2, the U-shape network can be divided into three parts: 1) the main body of the model, simultaneously bridged by an identity connection. 2) A *head block* and 3) a *tail block*, in which the latter two are designed for manipulating the input in spectral domain: $f_{head} : \mathbb{R}^{N_x \times Ny \times N_\lambda} \to \mathbb{R}^{N_x \times Ny \times N_\gamma}$, and f_{tail} does the inverse. A convolutional layer is used for this transformation.

3.2.1 NESTED RESIDUAL LEARNING

Besides the global residual learning, we exchange the convolutional modules and deconvolutional modules in plain U-Net with M cascaded identical Residual Blocks (ResBlock, RB). The feed-forward operation between ResBlocks is proceeded as

$$f_m = H_m(f_{m-1}), \tag{11}$$

where $H_m(\cdot)$ denotes the residual learning given by a CONV-ReLU-CONV structure with an identity connection, *i.e.*, $H_m(x) = \mathrm{CONV}(\mathrm{ReLU}(\mathrm{CONV}(x))) + x$, and f_{m-1} represents the output of the $(m-1)$th ResBlock. The whole stacking residual blocks could be represented as

$$f_M = H_M(H_{M-1}(\dots H_2(H_1(f_0))\dots)), \tag{12}$$

where f_0 is given as the output of the head block f_{head} in our model. Thus, we could treat all the stacking ResBlocks in equation 12 as a mapping function $f_M : \mathbb{R}^{N_x \times N_y \times N_\lambda} \rightarrow \mathbb{R}^{N_x \times N_y \times N_\lambda}$, and define our whole reconstruction model as

$$f_{res}(\boldsymbol{F}_Y) = f_{tail}(f_M(f_{head}(\boldsymbol{F}_Y)) + f_{head}(\boldsymbol{F}_Y)). \tag{13}$$

Pure residual learning He et al. (2016) releases the burden of mapping function by reformulating the learning objective $\boldsymbol{G}(x)$ as $\boldsymbol{G}(x) := F(x) + x$. We make a step forward by learning the "redisual of residual", namely nested residual learning. It refers to both the global skip connection that bypass the main body f_M and multiple skip connections that lie in residual blocks (RBs). Similar operation has been widely used in single image super resolution Lim et al. (2017) and image deblurring Gao et al. (2019), but unfortunately, not been well leveraged in top HSI reconstructive methods.

3.2.2 SPATIAL/SPECTRAL INVARIANCE

As shown in Tab 1, the performance of plain U-Net Ronneberger et al. (2015) for HSI reconstruction is sub-optimal, compared with state-of-the-art models. One underlying reason is the spatial contracting and expansive mechanism. In contracting path, Maxpooling operations lead to a lossy compression regarding spatial information, i.e., 1) pixels with lower intensity are discarded and 2) the location information of max-valued pixels are also missed. In expansive path, the Deconvolutional layers recover the spatial details in a learnable fashion. This paired operation actually raises the burden of learning. For both of the above reasons, we turn to create the mapping function (DCNN model) in a spatial invariant fashion.

To approximate the 3D signal in a rank minimization manner, previous researchers attempt to reduce the spectral channels somewhere in the model, which turns out to be a lossy compression operation and is avoided in our work. The plain U-Net tends to enlarge the spectral dimensions when reducing spatial size and do vice versa, which works well for the input image with single or few channels, i.e., gray image or RGB image, which however, is improper for the reconstructive inputs that are already packaged with considerable channels (i.e., in this work we use 28 channels). Therefore, in the proposed model, we firstly expanding the spectral channel from $N_\lambda = 28$ to $N_\gamma = 64$ and keeps it through the main body (spectral invariant). The underlying intention of augmentation is to increase the spectral-wise redundancy of intermediate embeddings.

3.2.3 TOWARD EFFICIENCY AND FLEXIBILITY OF PROPOSED METHOD

Although existing deep learning-based reconstructive models have achieved amazing performance, two problems still limit the deployment of them under practical scenarios.

The first problem is most existing models need heavy training time and inference time, which can be summarized as *inefficiency*. This can be solved by minimizing the model size (reducing the amount of trainable parameters). Although our proposed model contains less than 3% trainable parameters compared with the current best model TSA-Net Meng et al. (2020b), there still exists potential to further shrink the model size. We simplified our ResBlocks by removing the batch normalization Ioffe & Szegedy (2015) layer inside, leading to trainable parameter reduction. As compared in Tab. 2, the batch normalization actually makes no contribution on HSI reconstructive performance. Previous work in video SCI Cheng et al. (2020) came to the similar conclusion.

The other problem is that although existing E2E models have the potential to reconstruct the high resolution HSIs, they always fail under practical situations due to the excessive requirement of the

Table 1: PSNR (dB) and SSIM values by different algorithms on 10 scenes in the simulation dataset. Our approach is denoted by `v1`: SSI-ResU-Net w/o rescaling pair, `v2`: SSI-ResU-Net with 1 rescaling pair, and `v3`: SSI-ResU-Net with 2 rescaling pair. Notably, `v2` and `v3` are provided to further reduce the FLOPs of `v1` (see Table 2) while still exhibiting a competitive performance over seven state-of-the-art methods.

Scene	TwIST Bioucas-Dias & Figueiredo (2007)		GAP-TV Yuan (2016a)		DeSCI Liu et al. (2019)		U-net Ronneberger et al. (2015)		HSSP Wang et al. (2019)		λ-net Miao et al. (2019)		TSA-NET Meng et al. (2020b)		SSI-ResU-Net (v3)		SSI-ResU-Net (v2)		SSI-ResU-Net (v1)	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
1	24.81	0.730	25.13	0.724	27.15	0.794	28.28	0.822	31.07	0.852	30.82	0.880	31.26	0.887	32.85	0.906	33.16	0.909	**34.06**	**0.926**
2	19.99	0.632	20.67	0.630	22.26	0.694	24.06	0.777	26.30	0.798	26.30	0.846	26.88	0.855	28.61	0.850	29.08	0.852	**30.85**	**0.902**
3	21.14	0.764	23.19	0.757	26.56	0.877	26.02	0.857	29.00	0.875	29.42	0.916	30.03	0.921	31.27	0.903	32.19	0.902	**33.14**	**0.924**
4	30.30	0.874	35.13	0.870	39.00	0.965	36.33	0.877	38.24	0.926	37.37	0.962	39.90	0.964	39.42	0.954	39.81	0.943	**40.79**	**0.970**
5	21.68	0.688	22.31	0.674	24.80	0.778	25.51	0.795	27.98	0.827	27.84	0.866	28.89	0.878	29.93	0.911	30.43	0.911	**31.57**	**0.939**
6	22.16	0.660	22.90	0.635	23.55	0.753	27.97	0.794	29.16	0.823	30.69	0.886	31.30	0.895	32.81	0.927	33.22	0.927	**34.99**	**0.955**
7	17.71	0.694	17.98	0.670	20.03	0.772	21.15	0.799	24.11	0.851	24.20	0.875	25.16	0.887	26.26	0.839	26.69	0.838	**27.93**	**0.861**
8	22.39	0.682	23.00	0.624	20.29	0.740	26.83	0.796	27.94	0.831	28.86	0.880	29.69	0.887	30.87	0.913	31.50	0.918	**33.24**	**0.949**
9	21.43	0.729	23.36	0.717	23.98	0.818	26.13	0.804	29.14	0.822	29.32	0.902	30.03	0.903	31.74	0.909	31.69	0.910	**33.58**	**0.931**
10	22.87	0.595	23.70	0.551	25.94	0.666	25.07	0.710	26.44	0.740	27.66	0.843	28.32	0.848	29.84	0.893	29.98	0.893	**31.55**	**0.934**
Avg.	22.44	0.703	23.73	0.683	25.86	0.785	26.80	0.803	28.93	0.834	29.25	0.886	30.15	0.893	31.36	0.900	31.77	0.900	**33.17**	**0.929**

computational resources, considering the high floating point operations (FLOPs, *i.e.*, amount of additions and multiplications), i.e., TSA-Net takes up over 18GB for 1024×1024×28 HSI restoration and exceeds the burden of the general GPUs, which we summarize as *inflexibility*. Take the convolutional layer for example, the computation of FLOPs can be computed through following equation:

$$FLOPs = (2 \times ch_{in} \times \boldsymbol{K}^2 - 1) \times ch_{out} \times Height_{out} \times Width_{out}, \tag{14}$$

here ch_{in}, ch_{out} mean the input or output channels of a convolutional layer. $\boldsymbol{K}$ is the kernel size, $Height_{out}$ and $Width_{out}$ refer to the spatial sizes of the output feature map. Given a DCNN model, for the objective of minimizing the FLOPs, the simpliest solution is to reduce the spatial size of the intermediate embedding in the model. We achieve this objective via plugging in rescaling manipulations (*i.e.*, orange blocks in Fig. 2). A pair of rescaling manipulation includes a *downsampling* operation, which is a convolutional layer with stride 2, and an *upsampling* operation, which is a convolutional layer followed by *PixelShuffle* operation Shi et al. (2016). Spatial size will be doubled/halved by a single upsampling/downsampling operation. As drawn in Fig. 2 bottom-right, we put two pairs of rescaling manipulations in the model. The dashed box means the operations are adjustable, enabling a broad spectrum of image patch processing. Given an original input of spatial size 256×256, the embedding Λ is of 128×128 and embedding Γ is of 64×64. We explored three variants in total. The first version (`v1`), is the original model without rescaling manipulations. The second version (`v2`), contains one rescaling pair, which reduces the spatial size at the entry of main body and recovers when it ends (*i.e.*, in Fig. 2 bottom-right). The third version (`v3`), in which second rescaling pair is added in the main body of the model, with intermediate *RB* number k=8. Interestingly, experimental results below proves that all of proposed variants achieve state-of-the-art performance (Table 1) in simulation (in which `v1` is the best), also validate the superiority of proposed method on real data. In Fig. 6, we visualize real estimation of the second version (`v2`).

3.2.4 Training and Testing of SSI-ResU-Net

The model is trained to minimize a mean squared error (MSE) between the ground truth and output

$$\mathcal{L}_{MSE}(\Theta) = \frac{1}{N} \sum_{n=1}^{N} ||\widehat{\boldsymbol{F}}_n - \boldsymbol{F}_n||^2, \tag{15}$$

where $\widehat{\boldsymbol{F}} = f_{res}(\boldsymbol{F}_Y)$ given in equation 13 and N denotes the total N training samples.

For simulation experiment, synthetic hyperspectral images are sent into CASSI for reconstructing-aimed input initialization. Therefore, the 3D cube naturally becomes the ground truth. Training and testing data are separately abstracted from different datasets. For real-world HSI reconstruction, the measurements generated by a real CASSI system, combined with the same mask we previous used in simulation study, will be together fed into the learned model. Notably, the real-world captured measurements are disturbed by the noise introduced by the optical system. Therefore, we add Gaussian noise with standard deviation randomly generated from the range $[0, 0.05]$ to mimic this scenario.

4 Experiments

In this section, we evaluate the proposed method on both synthetic and real-image datasets. We design the empirical evaluations for answering the following questions:

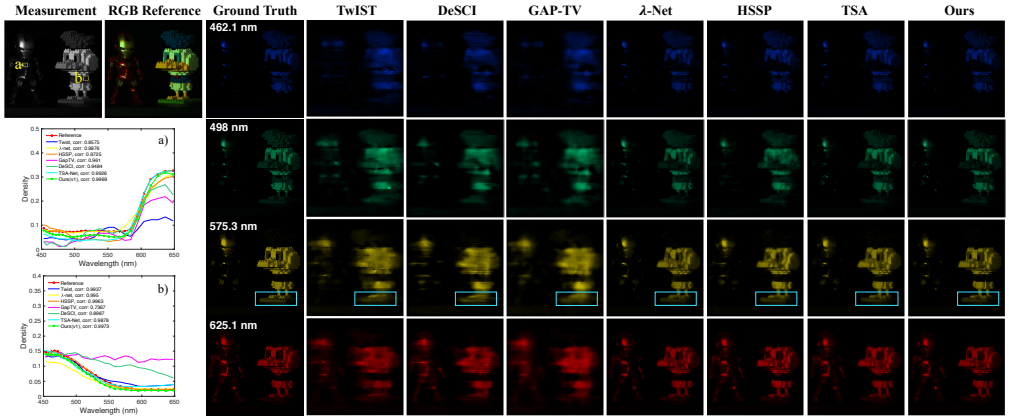


Figure 3: Comparison of reconstructed results for synthetic HSI on the *Ironman and Duck* scene. Six state-of-the-art methods and our proposed method (right column) are included. The RGB reference is shown to demonstrate the color (top-left). The density curves (bottom-left) are corresponding to the boxes marked in measurement. Our results recover the most details contained in real scene, *i.e.*, cyan boxes for 575.3nm wavelength. Also the spectral curves have the highest correlation to the reference (0.9969, 0.9973).

- *How does our approach compare against with the state-of-the-art on simulation data?* (Section 4.2)
- *What does the reconstruction look like by using our method on the real images?* (Section 4.3)
- *Can the proposed method achieve promising performance by extremely reducing the model size or lowering the computational cost?* (Section 4.4)

4.1 Experimental Settings

Dataset. For a fair comparison, we adopt the same 28 wavelengths for the HSI as in Meng et al. (2020b), distributed within the range of 450nm to 650nm obtained by spectral interpolation manipulation, and conduct experiments on the following two datasets: 1) Simulation Data and 2) Real HSI data.

Simulation Data. Both CAVE Park et al. (2007) synthetic dataset and KAIST Choi et al. (2017) synthetic dataset are applied in our simulation experiment. For the training set, we create 205 $1024\times1024\times28$ large image examplers from 30 $256\times256\times28$ images from CAVE dataset by randomly concatenating. To make the dataset as robust as we can, operations like rotation and rescaling are both used. Training samples of size $256\times256\times28$ will be further randomly cropped from the examplers. For testing set, ten different $256\times256\times28$ HSIs abstracted from KAIST dataset are applied, which will be leveraged to compare with other reconstruction methods.

Real HSI Reconstruction. We expand the previous training set by adding 37 HSI images in KAIST Choi et al. (2017) dataset, all of which will be cropped in of the shape $660\times660\times28$ to match the real-world measurement, which are abstracted by CASSI system developed in Meng et al. (2020b).

Compared Methods. We compare with seven state-of-the-art reconstruction algorithms, including TwIST Bioucas-Dias & Figueiredo (2007), GAP-TV Yuan (2016b), DeSCI Liu et al. (2019), U-Net Ronneberger et al. (2015), HSSP Wang et al. (2019), λ-net Miao et al. (2019) and TSA-Net Meng et al. (2020b), among which the most recent TSA-Net and λ-net methods have shown a promising performance in terms of simulation reconstruction. Two widely-used validation criteria, Peak Signal-to-Noise Racial (PSNR) and Structural SIMilarity (SSIM) Wang et al. (2004) are used for a quantitative comparison. The PSNR is computed by

$$PSNR_{ch} = 10\log_{10}(\frac{MAX_I^2}{MSE_{ch}}), \tag{16}$$

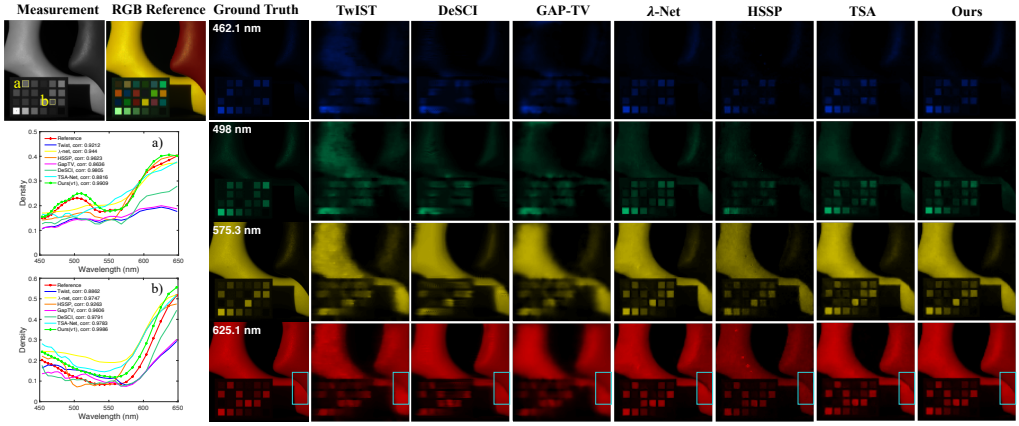


Figure 4: Comparison of reconstructed results for synthetic HSI (*Art* scene). Six state-of-the-art methods and our method (right-column) are included. The RGB reference is shown to demonstrate the color (Top-left). The spectral curves (bottom-left) are corresponding to the boxes denoted in measurement. As highlighted in cyan boxes on 625.1nm, our results are almost the same as the reference and contain no artifacts.

where $PSNR_{ch}$ means we compute the channel-wise PSNR value and average over all spectral channels. For a certain channel ch, MAX_I^2 denotes the maximum pixel value in ground truth image I. All computational values of PSNR in Table 1 follows this formula.

We implemented our model by PyTorch and trained it with the ADAM optimizer. We put 16 *Simplified ResBlock*s (*i.e.*, M=16) in main body. The learning rate was initialized as 4×10^{-4}, decreased by half every 50 epochs, for both simulation/real settings. We set batch size = 4 for the best validation performance. The training takes at most 16 hours with one NVIDIA TIATN RTX GPU on simulation data.

4.2 Experiments on Synthetic Data

As shown in Table 1, the proposed method (*i.e.*, our best model without rescaling, denoted as `SSI-ResU-Net(v1)`) has achieved the highest averaged PSNR and SSIM of 10 synthetic testing samples as 33.17 dB and 0.929, respectively, which is an striking breakthrough. Our method boosts 3.02 dB/0.036 compared with the previously best model, TSA-Net, in terms of PSNR/SSIM, and outperforms the λ-net by 3.92 dB/0.043, respectively. More importantly, we rank first in each of the testing samples evaluated by PSNR and SSIM. It is worthy noting that, although the other two versions of our model (*i.e.,* the proposed method with 1 or 2 rescaling pairs, termed as, `SSI-ResU-Net(v2)` and `SSI-ResU-Net(v3)`) are mainly designed for reducing the computational cost, they also exhibit the state-of-the-art performance compared with existing methods. The TSA-Net falls behind for 1.21 dB according to PSNR and 0.007 according to SSIM. Obviously, validated by three variants of our model, the proposed method demonstrated great stability and effectiveness.

Fig. 3 visualizes the reconstructed results between six compared methods and our model. Our reconstructed images contain the most details among four chosen spectral channels including 462.1nm, 498nm, 575.3nm and 625.1nm. As highlighted in the cyan box drawn in wavelength of 565.3nm (better visualization performance by zooming in), several cracks between the bumps around the feet of duck are obvious in our result, while these details are lost, instead, only blurred region can be seen in other results. We also plot the spectral density curves corresponding to two picked regions (box "a" and "b" in Fig. 3). The highest correlations (*i.e.*, 0.9969, 0.9975) and highly coincidence between our curves and reference proves the spectral-wise effectiveness of our model. Similarly, as shown by one more example in Fig. 4, our reconstructed results surpass all the other algorithms in terms of high frequency image detail recovery and spectral-wise restoration consistency. To clearly check the reconstructive ability of the proposed model (i.e., (`v1`)), we put RGB reconstructions of eight 256×256×28 estimated simulation results in Fig. 7.

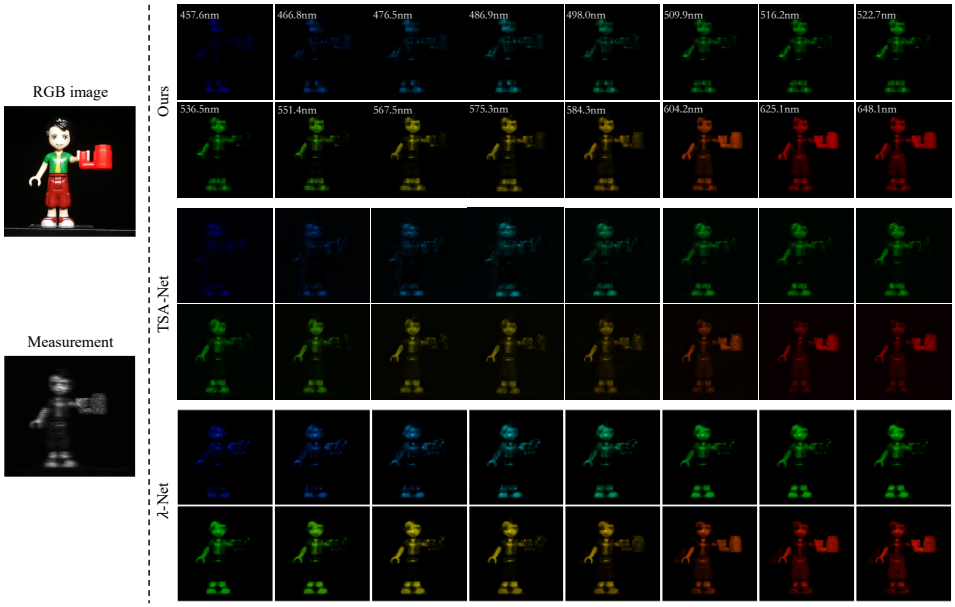


Figure 5: Reconstructed real hyperspectral images comparison between λ-net (bottom-right), TSA-Net (middle-right) and our method (top-right) from the real-captured CASSI measurement (bottom-left). The RGB reference is shown to demonstrate the color (Top-left). By comparison, our results give little distortion and artifact.

4.3 Experiments on Real Data

In addition to the achievement we have made on synthetic dataset, we validate the effectiveness of proposed method on hyperspectral images collected from real scenes. We show one example of a plant in Fig. 1 (a) by comparing our method with two best reconstructive methods, λ-net Miao et al. (2019) and TSA-Net Meng et al. (2020b), on 4 spectral channels. As can be seen, our method recovers the most textures. In Fig. 5, we also list 16 spectral channels (out of 28) of restored image (better visualization performance by zooming in). Taking the "face" of legoman for example, the reconstructed results of our method shows delicate details, which presents a sense of depth. While the output of λ-net for this region is totally distorted and that of TSA-Net is seriously corrupted.

We also reconstructed the real-world HSI model `v2` (*i.e.*, with one rescaling pair). Fig. 6 shows the reconstructive results of our model, TSA-Net and λ-net for three spectral channels in the right part. Our model provides a better brightness and details especially for purple and red flower, as circled in yellow boxes for 584.3nm's case. For 604.2nm, our reconstructed red flower looks more stereo than those reproduced by TSA-Net and λ-net. Though our reconstructed yellow flower is sub-optimal in 584.3nm, especially compared with λ-net, this is still a highly acceptable result, considering the shrinking feature maps were learned when the rescaling manipulation is applied. We plot the spectral density curves for two small regions (*i.e.*, box "a" and "b" in RGB reference). Our reproduced curve is most correlated to the reference, which is pre-determined by spectrometer, for which the spectral for the same color patch are assumed to be the same.

4.4 Efficiency Analysis

Model size. The ordinary performance improvement is obtained at the sacrifice of enlarged model size or increased model complexity. While we make breakthroughs in performance, we further compare the model size and floating-point operations (FLOPs) between three competitive deep learning-based methods and three variants of our proposed method. As shown in Table 2, our best-performance model (*i.e.*, the one without rescaling pairs) contains only 2.82% trainable parameters compared with TSA-Net, simultaneously reduces 96% and 98% trainable parameters when compared with U-net Ronneberger et al. (2015) and λ-net, respectively. The other two variants involve a little bit more trainable parameters due to the additional covolutional layer introduced by the rescaling pairs, for which the incremental in the model size can be neglected. With smaller storage space

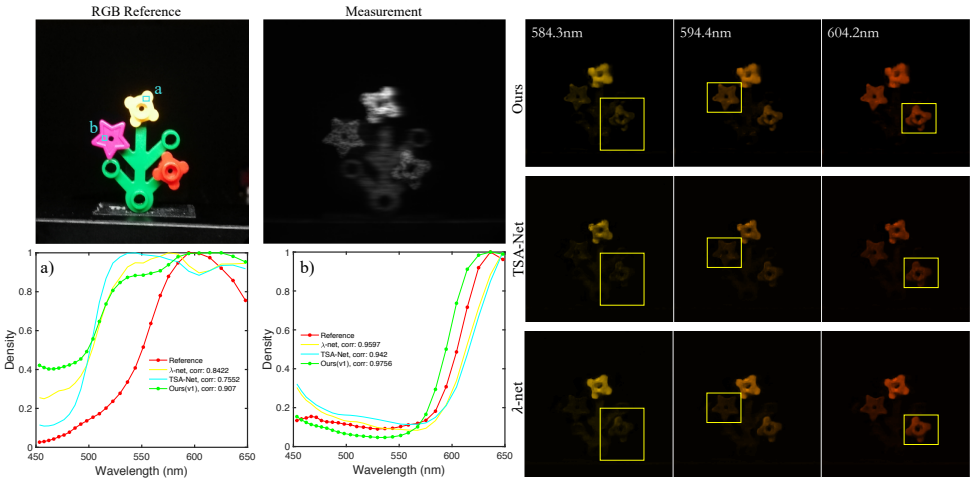


Figure 6: Reconstructed real HSI comparison between λ-net, TSA-Net and our proposed model `v2`, (*i.e.*, with one rescaling pair) from the real CASSI measurement (top-left).

Figure 7: RGB reconstruction of eight estimated simulation results.

Method	#parameters (M)	FLOPs (G)	PSNR (dB)
U-net Ronneberger et al. (2015)	31.32	58.99	26.80
λ-net Miao et al. (2019)	62.64	117.98	29.25
TSA-Net Meng et al. (2020b)	44.25	80.08	30.15
SSI-ResU-Net (v1)	1.25	81.84	33.17
SSI-ResU-Net (v2)	1.44	25.07	31.77
SSI-ResU-Net (v3)	1.62	18.57	31.36

Table 2: Analysis efficiency of the deep learning-based algorithms, with a spatial size of input as 256×256. The annotation `v1`: SSI-ResU-Net w/o rescaling pair, `v2`: SSI-ResU-Net with 1 rescaling pair, and `v3`: SSI-ResU-Net with 2 rescaling pair.

occupation, our models can be flexibly deployed, which potentially benefits the real-time reconstruction.

Floating-Point Operations (FLOPs). Given the input of spatial shape 256×256, we count the total amount of addition and multiplication for different deep learning-based model. As shown in Table 2, the FLOPs of the proposed model without rescaling pairs is 81.84G, which are slightly more than that of TSA-Net. The underlying intuition is that though the proposed model is rather small, the amount of operations conducted in the model is still formidable, which limits the capacity of processing large image patches. As shown in Table 2, two variants (*i.e.*, the bottom two models, `SSI-ResU-Net(v2)` and `SSI-ResU-Net(v3)`) mitigate this issue successfully. Typically, we reduce 69.4% FLOPs by using one rescaling pairs, which is less than 1/3 the FLOPs of TSA-Net. If we further add one more rescaling pair, the resulting FLOPs will be only 18.57G.

5 Conclusion

We have provided a simple yet highly efficient method specially-designed for hyperspectral imaging reconstruction in this paper. The proposed method outperformed existing methods quantitatively and qualitatively, by setting a new state-of-the-art performance with a negligible amount of model parameters. Without sacrificing the performance, we solve the problem of inefficiency and infeasibility of the neural network-based reconstructive methods. Our method can not only be solely referenced as a baseline, also conjointly served as a qualified backbone for future research in the HSI community.

REFERENCES

J.M. Bioucas-Dias and M.A.T. Figueiredo. A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration. *IEEE Transactions on Image Processing*, 16(12):2992–3004, December 2007. 1, 7, 8

Marcus Borengasser, William S Hungate, and Russell Watkins. *Hyperspectral remote sensing: principles and applications*. CRC press, 2007. 1

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

Inchang Choi, MH Kim, D Gutierrez, DS Jeon, and G Nam. High-quality hyperspectral reconstruction using a spectral prior. Technical report, 2017. 8

Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pp. 424–432. Springer, 2016. 3

D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, April 2006. 5

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. 5

Hongyun Gao, Xin Tao, Xiaoyong Shen, and Jiaya Jia. Dynamic scene deblurring with parameter selective sharing and nested skip connections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3848–3856, 2019. 6

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016. 6

Tao Huang, Weisheng Dong, Xin Yuan, Jinjian Wu, and Guangming Shi. Deep gaussian scale mixture prior for spectral compressive imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16216–16225, 2021. 3

Nabil Ibtehaz and M Sohel Rahman. Multiresunet: Rethinking the u-net architecture for multimodal biomedical image segmentation. *Neural Networks*, 121:74–87, 2020. 3

Vladimir Iglovikov and Alexey Shvets. Ternausnet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation. *arXiv preprint arXiv:1801.05746*, 2018. 3

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pp. 448–456. PMLR, 2015. 6

S. Jalali and X. Yuan. Snapshot compressed sensing: Performance bounds and algorithms. *IEEE Transactions on Information Theory*, 65(12):8005–8024, Dec 2019. ISSN 1557-9654. doi: 10.1109/TIT.2019.2940666. 5

Shirin Jalali and Xin Yuan. Compressive imaging via one-shot measurements. In *IEEE International Symposium on Information Theory (ISIT)*, 2018. 5

Min H Kim, Todd Alan Harvey, David S Kittle, Holly Rushmeier, Julie Dorsey, Richard O Prum, and David J Brady. 3d imaging spectroscopy for measuring hyperspectral patterns on solid objects. *ACM Transactions on Graphics (TOG)*, 31(4):1–11, 2012. 1

Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 136–144, 2017. 6

Yang Liu, Xin Yuan, Jinli Suo, David Brady, and Qionghai Dai. Rank minimization for snapshot compressive imaging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(12): 2990–3006, Dec 2019. 1, 7, 8

Patrick Llull, Xuejun Liao, Xin Yuan, Jianbo Yang, David Kittle, Lawrence Carin, Guillermo Sapiro, and David J Brady. Coded aperture compressive temporal imaging. *Optics Express*, 21(9):10526–10545, 2013. ISSN 1094-4087 (Electronic) 1094-4087 (Linking). doi: 10.1364/OE.21.010526. 1

Guolan Lu and Baowei Fei. Medical hyperspectral imaging: a review. *Journal of biomedical optics*, 19(1):010901, 2014. 1

Jiawei Ma, Xiaoyang Liu, Zheng Shou, and Xin Yuan. Deep tensor admm-net for snapshot compressive imaging. In *IEEE/CVF Conference on Computer Vision (ICCV)*, 2019. 2

Farid Melgani and Lorenzo Bruzzone. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Transactions on geoscience and remote sensing*, 42(8):1778–1790, 2004. 1

Ziyi Meng, Shirin Jalali, and Xin Yuan. Gap-net for snapshot compressive imaging. *arXiv preprint arXiv:2012.08364*, 2020a. 3

Ziyi Meng, Jiawei Ma, and Xin Yuan. End-to-end low cost compressive spectral imaging with spatial-spectral self-attention. In *European Conference on Computer Vision (ECCV)*, August 2020b. 1, 2, 3, 4, 6, 7, 8, 10, 11

Ziyi Meng, Mu Qiao, Jiawei Ma, Zhenming Yu, Kun Xu, and Xin Yuan. Snapshot multispectral endomicroscopy. *Optics Letters*, 45(14):3897–3900, 2020c. 1, 3

Xin Miao, Xin Yuan, Yunchen Pu, and Vassilis Athitsos. λ-net: Reconstruct hyperspectral images from a snapshot measurement. In *IEEE/CVF Conference on Computer Vision (ICCV)*, 2019. 1, 2, 3, 7, 8, 10, 11

Ozan Oktay, Jo Schlemper, Loic Le Folgoc, Matthew Lee, Mattias Heinrich, Kazunari Misawa, Kensaku Mori, Steven McDonagh, Nils Y Hammerla, Bernhard Kainz, et al. Attention u-net: Learning where to look for the pancreas. *arXiv preprint arXiv:1804.03999*, 2018. 3

J. Park, M. Lee, M. D. Grossberg, and S. K. Nayar. Multispectral Imaging Using Multiplexed Illumination. In *IEEE International Conference on Computer Vision (ICCV)*, Oct 2007. 8

Antonio Plaza, Jon Atli Benediktsson, Joseph W Boardman, Jason Brazile, Lorenzo Bruzzone, Gustavo Camps-Valls, Jocelyn Chanussot, Mathieu Fauvel, Paolo Gamba, Anthony Gualtieri, et al. Recent advances in techniques for hyperspectral image processing. *Remote sensing of environment*, 113:S110–S122, 2009. 1

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015. 2, 3, 6, 7, 8, 10, 11

Swalpa Kumar Roy, Subhrasankar Chatterjee, Siddhartha Bhattacharyya, Bidyut B Chaudhuri, and Jan Platoš. Lightweight spectral–spatial squeeze-and-excitation residual bag-of-features learning for hyperspectral classification. *IEEE Transactions on Geoscience and Remote Sensing*, 58(8): 5277–5290, 2020. 4

Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1874–1883, 2016. 7

Chao Tao, Hongbo Pan, Yansheng Li, and Zhengrou Zou. Unsupervised spectral–spatial feature learning with stacked sparse autoencoder for hyperspectral imagery classification. *IEEE Geoscience and remote sensing letters*, 12(12):2438–2442, 2015. 4

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

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

L. Wang, Z. Xiong, G. Shi, F. Wu, and W. Zeng. Adaptive nonlocal sparse representation for dual-camera compressive hyperspectral imaging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(10):2104–2111, Oct 2017. ISSN 0162-8828. doi: 10.1109/TPAMI.2016.2621050. 1

L. Wang, C. Sun, Y. Fu, M. H. Kim, and H. Huang. Hyperspectral image reconstruction using a deep spatial-spectral prior. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8024–8033, June 2019. doi: 10.1109/CVPR.2019.00822. 3, 7, 8

Lizhi Wang, Tao Zhang, Ying Fu, and Hua Huang. Hyperreconnet: Joint coded aperture optimization and image reconstruction for compressive hyperspectral imaging. *IEEE Transactions on Image Processing*, 28(5):2257–2270, May 2019. ISSN 1057-7149. doi: 10.1109/TIP.2018.2884076. 1, 3

Lizhi Wang, Chen Sun, Maoqing Zhang, Ying Fu, and Hua Huang. Dnu: Deep non-local unrolling for computational spectral imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 3

Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. 8

Yang Xu, Zebin Wu, Jun Li, Antonio Plaza, and Zhihui Wei. Anomaly detection in hyperspectral images based on low-rank and sparse representation. *IEEE Transactions on Geoscience and Remote Sensing*, 54(4):1990–2000, 2015. 1

X. Yuan, D. J. Brady, and A. K. Katsaggelos. Snapshot compressive imaging: Theory, algorithms, and applications. *IEEE Signal Processing Magazine*, 38(2):65–88, 2021. 1, 5

Xin Yuan. Generalized alternating projection based total variation minimization for compressive sensing. In *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 2539–2543, Sept 2016a. 1, 7

Xin Yuan. Compressive dynamic range imaging via Bayesian shrinkage dictionary learning. *Optical Engineering*, 55(12):123110, 2016b. 8

Xin Yuan, Tsung-Han Tsai, Ruoyu Zhu, Patrick Llull, David Brady, and Lawrence Carin. Compressive hyperspectral imaging with side information. *IEEE Journal of Selected Topics in Signal Processing*, 9(6):964–976, September 2015. 1

Yuan Yuan, Xiangtao Zheng, and Xiaoqiang Lu. Hyperspectral image superresolution by transfer learning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(5):1963–1974, 2017. 1

Siming Zheng, Yang Liu, Ziyi Meng, Mu Qiao, Zhishen Tong, Xiaoyu Yang, Shensheng Han, and Xin Yuan. Deep plug-and-play priors for spectral snapshot compressive imaging. *Photon. Res.*, 9 (2):B18–B29, Feb 2021. 3