Uncertainty-Driven Spectral Compressive Imaging with Spatial-Frequency Transformer

Lintao Peng[1], Siyu Xie[1], and Liheng Bian[1]

Beijing Institute of Technology, Beijing, 100081, China. bian@bit.edu.cn

Abstract. Recently, learning-based Hyperspectral image (HSI) reconstruction methods have demonstrated promising performance. However, existing learning-based methods still face two issues. 1) They rarely consider both the spatial sparsity and inter-spectral similarity priors of HSI. 2) They treat all image regions equally, ignoring that texture-rich and edge regions are more difficult to reconstruct than smooth regions. To address these issues, we propose an uncertainty-driven HSI reconstruction method termed Specformer. Specifically, we first introduce a frequency-wise self-attention (FWSA) module, and combine it with a spatial-wise local-window self-attention (LWSA) module in parallel to form a Spatial-Frequency (SF) block. LWSA can guide the network to focus on the regions with dense spectral information, and FWSA can capture the inter-spectral similarity. Parallel design helps the network to model cross-window connections, and expand its receptive fields while maintaining linear complexity. We use SF-block as the main building block in a multi-scale U-shape network to form our Specformer. In addition, we introduce an uncertainty-driven loss function, which can reinforce the network's attention to the challenging regions with rich textures and edges. Experiments on simulated and real HSI datasets show that our Specformer outperforms state-of-the-art methods with lower computational and memory costs. The code is available at https://github.com/bianlab/Specformer.

Keywords: Hyperspectral Imaging · Spatial-Frequency Transformer · Uncertainty-Driven Learning

1 Introduction

Compared with RGB images, hyperspectral images (HSI) have more spectral bands, which makes them able to store richer spectral information and delineate more detailed characteristics of the target scene. Benefiting from this property, HSIs have been widely used in multiple computer vision tasks, such as object detection [22, 40], remote sensing [1, 33] and medical image processing [31, 36]. Conventional hyperspectral imaging systems scan the scene along the spatial or spectral dimension to obtain the HSI cubes. This scanning strategy is time-consuming, making it unsuitable for capturing and measuring dynamic scenes. Compared with traditional push-broom scanning spectral imaging techniques,

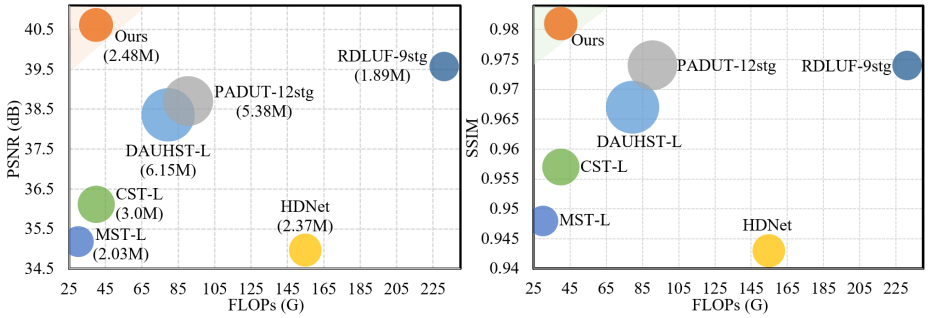

Fig. 1: PSNR-Params-FLOPs and SSIM-Params-FLOPs comparisons with exsiting HSI reconstruction methods. The circle radius represents the parameter numbers of the model. The reported Specformer technique outperforms state-of-the-art methods while requiring fewer FLOPs and Params.

snapshot compressive imaging (SCI) technology only requires one time of exposure to obtain a complete HSI cube. These SCI systems compress both spatial and spectral information into a single 2D measurement [52], and then use algorithms to reconstruct an HSI data cube [6, 13, 30, 47, 48]. The Coded Aperture Snapshot Spectral Imaging (CASSI) [47] technique stands out as a representative SCI technique.

In recent years, researchers have proposed multiple reconstruction algorithms to reconstruct 3D HSI cubes from 2D measurements. Traditional model-based methods use handcrafted priors such as sparsity [23, 27, 47], total variability [49, 51], and non-local similarity [28, 50, 53] to guide the reconstruction process. However, these methods rely on manually tuned parameters and often require different parameters for different scenes, which leads to poor generality and slow reconstruction speed. Compared to model-based methods, CNN-based HSI reconstruction methods [20, 34, 35, 37, 38] do not require manual parameter tuning, and produce improvements on generalization performance and reconstruction speed. However, CNN-based methods show limitations in capturing non-local self-similarity and long-range dependencies of HSI, which results in unsatisfactory HSI reconstruction quality.

Recently, Transformer [46] has shown promising performance in image processing. The self-attention mechanism in Transformer can model long-range dependencies and non-local similarities. These advantages offer the possibility to address the shortcomings of CNN-based methods. Based on the Transformer technique, researchers have reported performance-leading HSI reconstruction methods such as MST [3], CST [2], DAUHST [4], and PADUT [25]. However, the existing Transformer-based methods still face the following issues. **First**, in the global attention Transformer, the computational complexity is quadratic to the spatial size. This burden is non-trivial and sometimes unaffordable. The local-window self-attention (LWSA) [29] module can effectively reduce computational complexity, but the receptive field of LWSA module is quite limited.

Second, Transformer-based methods often ignore the spatial sparsity of HSI. These methods intensively collect all tokens and calculate global self-attention, causing a lot of computing resources wasting in areas with sparse spectral information. **Third**, the original Transformer method [12] learns to capture the long-range dependencies spatially, but the representations of HSIs are spectrally highly self-similar. In this case, the inter-spectral similarities are not well modeled.

To address these issues, we propose an uncertainty-driven HSI reconstruction method termed Specformer. Specifically, inspired by the spatial sparsity and inter-spectral similarity nature of HSIs, we first introduce frequency-wise self-attention (FWSA), a conceptually simple but computationally efficient architecture. It consists of the fast Fourier transform (FFT) [39], the learnable global filter, and the inverse fast Fourier transform (IFFT) [45]. FWSA module can calculate self-attention along the spectral dimension with linear complexity, modeling the inter-spectral long-distance dependencies and capturing the inter-spectral similarity of HSI. Next, we use a parallel design to combine it with the spatial-wise LWSA module to form a basic block termed spatial-frequency (SF) block. LWSA module can model the spatial sparsity and guide the network to focus on the spatial regions with dense spectral information. Parallel design helps the SF-block to model cross-window connections, and expand its receptive field while maintaining linear complexity. We insert the SF-block as the main building block in a U-shape architecture [44] to form our Specformer. In addition, considering that texture-rich and edge regions in HSIs are more difficult to reconstruct, we introduce an uncertainty-driven self-adaptive loss function to reinforce the network's attention on those regions, thus improving the HSI reconstruction quality. The contribution of this paper is summarised as follows,

- We propose a novel Specformer technique for HSI reconstruction. To the best of our knowledge, it is the first attempt to embed both the spatial sparsity and inter-spectral similarity of HSIs into learning-based reconstruction.
- We introduce a novel self-attention module termed FWSA, and combine it with the LWSA module in a parallel design to form an SF-block. It can model the spatial sparsity and inter-spectral similarity of HSIs.
- We introduce an uncertainty-driven self-adaptive loss to enhance the HSI reconstruction quality in texture-rich and edge regions.
- Our Specformer outperforms state-of-the-art methods in both quantitative evaluation and visual comparison, with lower computational and memory costs.

2 Related work

2.1 HSI Reconstruction

Traditional model-based methods [23, 27, 47, 49, 51] recover 3D HSI cubes from 2D measurements based on hand-crafted priors. For example, Ref. [15] solved the sparse HSI reconstruction problem using the gradient projection algorithm

based on the spatial sparse prior of HSI. In Ref. [51], the nonlocal self-similarity and low-rank properties of HSIs have been exploited to solve HSI reconstruction problems. These model-based methods rely on hand-crafted parameters, and are difficult to adapt to different scenes, leading to poor generalization ability and slow recovery speed.

Compared with model-based methods, CNN-based methods show improvements in generalization performance and reconstruction speed. The CNN-based techniques can be categorized into end-to-end (E2E) methods, deep unfolding methods, and plag-and-paly (PnP) methods. The E2E methods [16, 35, 38] aim to learn a mapping function from 2D measurements to 3D HSI cubes. The deep unfolding methods [11, 20, 32, 34] employ multi-stage CNNs trained to map the measurements into the desired signal. Each stage contains two parts, *i,e* , linear projection and passing the signal through a CNN functioning as a denoiser. The PnP methods [7, 43] insert the pre-trained CNN denoiser into a model-based optimization framework for HSI reconstruction. Despite of the developments, CNN-based methods still have limitations in capturing long-distance dependencies and modeling non-local similarities.

Recently, the Global Vision Transformer has achieved great success in image classification [12]. However, for the dense image processing tasks such as HSI reconstruction, the computational complexity of the Transformer technique is quadratic with the image size, making it unable to be directly applied for HSI reconstruction. Moreover, the existing Transformer-based methods [5, 19] generally ignore the spatial sparsity of HSI, and intensively collect all tokens and calculate global self-attention, causing a lot of computing resources waste in the areas with sparse spectral information. The MST [3] method circumvents the computational complexity and spatial sparsity problems by calculating self-attention in the spectral domain, but this makes it unable to effectively model spatial-wise local and non-local features. The CST [2] technique embeds the HSI spatial sparsity into the learning process through a coarse-to-fine learning scheme, which effectively reduces the computational complexity of the Transformer, but ignores the inter-spectral similarity.

2.2 Uncertainty-driven Loss

For the HSI reconstruction task, texture-rich and edge regions are more difficult to reconstruct compared to smooth regions, and the reconstruction quality of these regions is decisive for the final reconstruction quality. In previous HSI reconstruction methods [2, 3, 5], researchers tend to improve the reconstruction quality by designing deeper, larger, and more complex networks, where all pixels are still treated equally. However, treating each pixel equally during the training process is not the optimal choice for the HSI reconstruction task. Intuitively, we need a spatially self-adaptive loss function.

Recently, the uncertainty loss function [8, 17, 21] has attracted certain attention. The uncertainty in deep learning can be roughly divided into two categories [10]. Epistemic/model uncertainty describes how much the model is uncertain about its predictions. Another type is aleatoric/data uncertainty which

refers to noise inherent in observation data. The GRAM [24] technique analyses the effect of aleatoric/data uncertainty on image reconstruction. By decreasing the loss attenuation of large variance pixels, GRAM achieves better results than directly applying the above uncertainty loss to image enhancement. In these tasks, pixels with a high degree of uncertainty are considered unreliable pixels that will suffer loss attenuation. However, this contradicts the intuition that the regions with rich textures and edges should be given priority in the HSI reconstruction task. In this regard, different from the above methods, we propose a novel uncertainty-driven spatially self-adaptive loss function, which can assign larger training weights to the texture-rich and edge regions of HSI, thus improving the HSI reconstruction performance.

2.3 CASSI Model

CASSI [47] is a mature and widely used spectral imaging technology. All experiments in this paper are based on CASSI. Figure 2(c) shows the principle of a single-dispenser CASSI. We denote the 3D HSI cube as $\mathbf{F} \in \mathbb{R}^{H \times W \times N_\lambda}$, where H, W, and N_λ refer to the HSI's height, width, and number of wavelengths, respectively. $\mathbf{F}$ is first collected by the objective lens and spatially encoded along the channel dimension by a coded aperture $\mathbf{M}^* \in \mathbb{R}^{H \times W}$, which is denoted as

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

Among them, $\mathbf{F}'$ represents the signal modulated by the coded aperture, $n_\lambda \in [1, \ldots, N_\lambda]$ represents different spectral wavelengths, and $\odot$ represents element-wise multiplication. The $\mathbf{F}'$ passes through the disperser and becomes tilted, which can be considered as sheared along the y-axis. Assuming λ_c is the reference wavelength, then the dispersion can be formulated as

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

where $\mathbf{F}'' \in R^{H \times (W + d(N_\lambda - 1)) \times N_\lambda}$ is the signal after dispersion, d refers to the step of spatial shifting, (u, v) locates the coordinate on the sensing detector, λ_n represents the wavelength of the n_λ-th channel, and $d(\lambda_n - \lambda_c)$ refers to the spatial shifting offset of the n_λ-th channel on $\mathbf{F}''$. Eventually, the data cube is compressed into a 2D measurement $\mathbf{Y} \in \mathbb{R}^{H \times (W + d(N_\lambda - 1))}$ by integrating all the channels as

$$\mathbf{Y} = \sum_{n_\lambda = 1}^{N_\lambda} \mathbf{F}''(:,:,n_\lambda) + \mathbf{G}, \tag{3}$$

where $\mathbf{G} \in \mathbb{R}^{H \times (W + d(N_\lambda - 1))}$ represents the random noise during the imaging process. The core task of HSI reconstruction is to recover the 3D HSI cube $\mathbf{F}$ from the 2D measurement $\mathbf{Y}$.

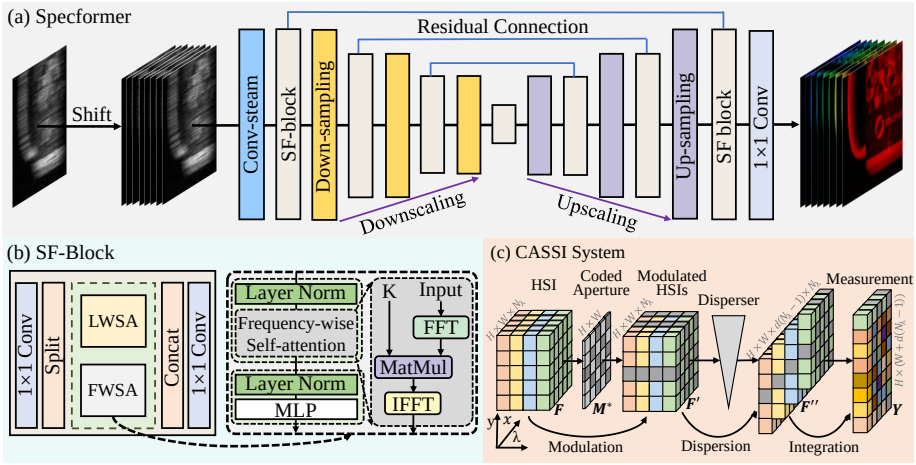

Fig. 2: (a) The overall structure of Specformer. (b) Detailed structure of the Spatial-Frequency (SF) block. We use a parallel design to combine the local-window self-attention (LWSA) module with the frequency-wise self-attention (FWSA) module to form the SF-block. LWSA can guide the network to focus on the regions with dense spectral information. FWSA can capture the inter-spectral similarity. It consists of the fast Fourier transform (FFT) [39], the learnable global filter $\mathbf{K}$ and the inverse fast Fourier transform (IFFT) [45]. Parallel design helps the specformer to model cross-window connections, and enlarge the receptive fields while maintaining linear complexity. (c) A schematic diagram of the CASSI model.

3 Method

3.1 Overall Reconstruction Architecture

The overall architecture of the reported Specformer technique is shown in Fig. 2(a), which consists of an encoder and a decoder built based on the SF-block. First, we reverse the dispersion process and shift back the measurement to obtain the initialized signal $\mathbf{H} \in \mathbb{R}^{H \times W \times N_\lambda}$ as

$$\mathbf{H}\left(\mathrm{x}, \mathrm{y}, n_\lambda\right) = \mathbf{Y}(\mathrm{x}, \mathrm{y} - \mathrm{d}(\lambda_n - \lambda_c)). \tag{4}$$

Next, we send $\mathbf{H}$ into Specformer. In Specformer, the input feature $\mathbf{H}$ is first processed by the conv-steam layer to convert the number of channels into C (we set C to be 32 in this work), and then obtain the preprocessing feature $\mathbf{X_0} \in \mathbb{R}^{H \times W \times C}$. Then, $\mathbf{X_0}$ is fed into the Specformer's encoder. There are four scales of encoding modules in the encoder, each of which contains several SF-blocks and a downsampling layer. Consequently, the output feature of the i-th stage of the encoder is denoted as $\mathbf{X_i^e} \in \mathbb{R}^{\frac{H}{2^i} * \frac{W}{2^i} * 2^i C}$. Except for being input to the next encoding module, the output of the i-th encoding module is directly input to the decoding module with the same scale through residual connections. The decoder also contains four scales of decoding modules, each of which contains

several SF-blocks and an upsampling layer. Similarly, the output of the i-th stage of the decoder is denoted as $\mathbf{X_i^d} \in R^{\frac{H}{2^i} * \frac{W}{2^i} * 2^i C}$. The final output features of the decoder are fed into a 1×1 convolutional layer to convert the number of channels into N_λ, and the final reconstruction result is $\mathbf{H'} \in R^{H*W*N_\lambda}$.

3.2 Spatial-Frequency (SF) block

The detailed SF-block structure is shown in Fig. 2(b). Assuming that the inputs of SF-block are feature maps $\mathbf{X_{in}} \in R^{\frac{H}{2^i} * \frac{W}{2^i} * 2^i C}$ at different scales. To be specific, for an input feature map $\mathbf{X_{in}}$, it is first passed through a 1×1 convolution and split evenly into two feature maps $\mathbf{X_1}$ and $\mathbf{X_2}$ as

$$\mathbf{X_1}, \mathbf{X_2} = \text{Split}(\text{Conv1} \times 1(\mathbf{X_{in}})). \tag{5}$$

Next, $\mathbf{X_1}$ is fed into the LWSA module for further processing. In the LWSA module, the feature map $\mathbf{X_1}$ is linearly mapped to generate a one-dimensional feature sequence $\mathbf{S_1} \in \mathbb{R}^{2^i C * \ d_i} (d = \frac{HW}{2^{2i}})$, which is then multiplied by the learnable weight matrices $W_Q \in \mathbb{R}^{2^i C * \ d_i}$, $W_K \in \mathbb{R}^{2^i C * \ d_i}$, and $W_V \in \mathbb{R}^{2^i C * \ d_i} (d = \frac{HW}{2^{2i}})$ to generate $\mathbf{Q}$, $\mathbf{K}$ and $\mathbf{V}$, respectively, after layer normalization. The above calculation process is denoted as

$$\mathbf{Q} = \mathbf{S_1} W_Q; \mathbf{K} = \mathbf{S_1} W_K; \mathbf{V} = \mathbf{S_1} W_V. \tag{6}$$

After obtaining $\mathbf{Q}$, $\mathbf{K}$ and $\mathbf{V}$, we implement the following calculation based on the self-attentive mechanism as

$$\mathbf{Y_1^{'}} = \text{SoftMax}(\text{IN}(\frac{\mathbf{Q^T K}}{\sqrt[2]{2^i C}}))\mathbf{V}, \tag{7}$$

where IN represents the instance normalization. The 1D feature sequence $\mathbf{Y_1^{'}}$ is then resized into 2D features $\mathbf{Y_1} \in \mathbb{R}^{\frac{H}{2^i} * \frac{W}{2^i} * 2^i C}$ using feature remapping.

Similarly, $\mathbf{X_2}$ is sent to the FWSA module. As shown in Fig. 2(b), in the FWSA module, we propose to use a global learnable filter $\mathbf{K}$ as an alternative to the self-attention mechanism to interchange information globally among the Fourier domain tokens. For the input feature $\mathbf{X_2}$, we first perform layer normalization (LN), and then use 2D FFT [26] to convert $\mathbf{X_2}$ to the Fourier domain as

$$\mathbf{X_F} = \mathcal{F}(\text{LN}(\mathbf{X_2})), \tag{8}$$

where $\mathcal{F}(\cdot)$ denotes the 2D FFT operation. The output feature $\mathbf{X_F}$ represents the Fourier spectrum of $\mathbf{X_2}$. We can then modulate the spectrum by multiplying a filter $\mathbf{K} \in \mathbb{R}^{\frac{H}{2^i} * \frac{W}{2^i} * 2^i C}$ to $\mathbf{X_F}$ as

$$\mathbf{Y_F} = \mathbf{K} \odot \mathbf{X_F}, \tag{9}$$

where $\odot$ is element-wise multiplication. The filter $\mathbf{K}$ is called the global filter since it has the same dimension as $\mathbf{X_F}$, which represents a learnable frequency

filter for different hidden dimensions [42]. Finally, we adopt the inverse FFT (IFFT) operation [45] to transform the modulated spectrum $\mathbf{Y_F}$ back to the spatial domain and update the tokens as

$$\mathbf{Y_2} = \mathcal{F}^{-1}(\mathbf{Y_F}), \tag{10}$$

where $\mathcal{F}^{-1}(\cdot)$ denotes the 2D IFFT.

Before concatenating, $\mathbf{Y_1}$ and $\mathbf{Y_2}$ need to be processed by a feed-forward layer respectively, which consists of an LN layer and an MLP layer [46]. Finally, $\mathbf{Y_1}$ and $\mathbf{Y_2}$ are concatenated as the input of a 1×1 convolution which has a residual connection with the input $\mathbf{X_{in}}$. As such, the final output of SF-block is given by

$$\mathbf{X_{out}} = \text{Conv}1 \times 1(\text{Concat}(\mathbf{Y_1}, \mathbf{Y_2})) + \mathbf{X_{in}}. \tag{11}$$

3.3 Uncertainty-Driven Loss

To reinforce the network's attention on the texture-rich and edge regions, as shown in Fig. 3, we divide the training of the network into two stages. In the first stage, the network estimates both the HSI cube and the uncertainty map. In the second stage, the uncertainty values are used to generate a spatially adaptive loss to guide the network to prioritize the pixels in the regions with rich textures and edges. To better quantify the arbitrary uncertainty in HSI reconstruction, as shown in Fig. 3, we use $\boldsymbol{x_i}, \boldsymbol{y_i}$ to denote the measurement and the corresponding ground truth, respectively. Let $f(.)$ denote an arbitrary HSI reconstruction network, and the aleatoric uncertainty is denoted by an additive term $\boldsymbol{\theta_i}$. The overall HSI reconstruction model can be formulated as

$$\boldsymbol{y_i} = f(\boldsymbol{x_i}) + \varepsilon \boldsymbol{\theta_i}, \tag{12}$$

where ε represents the Laplace distribution with zero-mean and unit-variance.

For a given input measurement $\boldsymbol{x_i}$ and corresponding HSI $\boldsymbol{y_i}$, a Laplace distribution is assumed for characterizing the likelihood function as

$$\mathrm{p}\left(\boldsymbol{y_i}, \boldsymbol{\theta_i} \mid \boldsymbol{x_i}\right) = \frac{1}{2\boldsymbol{\theta_i}} \exp\left(-\frac{\left\|\boldsymbol{y_i} - f\left(\boldsymbol{x_i}\right)\right\|_1}{\boldsymbol{\theta_i}}\right), \tag{13}$$

where $f(\boldsymbol{x_i})$ denotes the reconstruction results, and $\boldsymbol{\theta_i}$ denotes the uncertainty (variance) which are learned by the network. Then, the log-likelihood can be formulated as

$$\ln \mathrm{p}\left(\boldsymbol{y_i}, \boldsymbol{\theta_i} \mid \boldsymbol{x_i}\right) = -\frac{\left\|\boldsymbol{y_i} - f\left(\boldsymbol{x_i}\right)\right\|_1}{\boldsymbol{\theta_i}} - \ln \boldsymbol{\theta_i} - \ln 2. \tag{14}$$

For numerical stability, we train the Specformer network to estimate the uncertainty variance $\boldsymbol{s_i} = \ln(\boldsymbol{\theta_i})$ as shown in Fig. 3. Finally, the maximum likelihood estimate of Eq. 14 can be reformulated as minimizing the following loss function to estimate the uncertainty in the reconstruction process

$$\text{Loss}_U = \frac{1}{N} \sum_{i=1}^{N} \exp\left(-\boldsymbol{s_i}\right) \left\|\boldsymbol{y_i} - f\left(\boldsymbol{x_i}\right)\right\|_1 + \boldsymbol{s_i}. \tag{15}$$

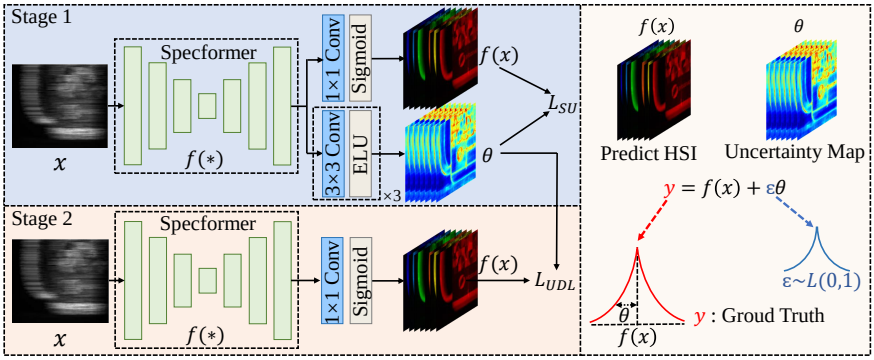

Fig. 3: The overview of the two-stage training strategy. The uncertainty estimation $\boldsymbol{\theta}$ serves as the bridge connecting two steps, i.e., it is the output of the first stage, and is passed to the second stage as the guidance required for calculating Loss_{UDL}.

The loss function Loss_U includes two terms. The first term is associated with reconstruction fidelity, and the second one prevents the network from predicting infinite uncertainty for all pixels. Based on the spatial sparsity nature of HSI and its uncertainty map, we propose to impose Jeffrey's prior [14] $p(\boldsymbol{w}) \propto \frac{1}{w}$ on uncertainty $\boldsymbol{\theta_i}$ as

$$p(\boldsymbol{y_i}, \boldsymbol{\theta_i} | \boldsymbol{x_i}) = p(\boldsymbol{y_i} | \boldsymbol{x_i}, \boldsymbol{\theta_i}) p(\boldsymbol{\theta_i}) \propto \frac{1}{2{\boldsymbol{\theta_i}}^2} \exp(-\frac{||\boldsymbol{y_i} - f(\boldsymbol{x_i})||_1}{\boldsymbol{\theta_i}}). \tag{16}$$

Then the log likelihood and loss function Loss_{SU}(SU represents sparse uncertainty) can be separately formulated as

$$\ln p\left(\boldsymbol{y_i} \mid \boldsymbol{x_i}\right) = -\frac{\left\|\boldsymbol{y_i} - f\left(\boldsymbol{x_i}\right)\right\|_1}{\boldsymbol{\theta_i}} - 2\ln \boldsymbol{\theta_i} - \ln 2, \tag{17}$$

$$\text{Loss}_{SU} = \frac{1}{N} \sum_{i=1}^{N} \exp\left(-\boldsymbol{s_i}\right) \left\|\boldsymbol{y_i} - f\left(\boldsymbol{x_i}\right)\right\|_1 + 2\boldsymbol{s_i}. \tag{18}$$

The above shows the loss function of the first training stage. When the Loss_{SU} converges, the trained Specformer network can be used to estimate the reconstruction uncertainty. Then, we construct a monotonically increasing function $\hat{\boldsymbol{s_i}} = \ln(1 + e^{\boldsymbol{s_i}})$ to prioritize the uncertainty values, and then use the ranked uncertainty as spatially adaptive weight to multiply with the HSI reconstruction loss function in the second stage as

$$\text{Loss}_{UDL} = \frac{1}{N} \sum_{i=1}^{N} \hat{\boldsymbol{s_l}} \left(\text{Loss}_{L_1} + \text{Loss}_{SSIM}\right), \tag{19}$$

where Loss_{L_1} and Loss_{SSIM} denote the Mean Absolute Error (MAE) loss and Structure Similarity Index Measure (SSIM) loss [18], respectively. In Loss_{UDL}, the texture and edge pixels with higher uncertainty tend to have greater weights than smooth regions.

4 Experiments

4.1 Experiment Setup

Datasets. For simulation comparison, we employed the CAVE [41] and the KAIST [9] dataset. The CAVE dataset consists of 32 HSIs with a spatial size of 512×512 pixels. The KAIST dataset contains 30 HSIs of spatial size 2704×3376. Following the settings of DGSMP [20], 28 wavelengths from 450nm to 650nm were derived by spectral interpolation. The CAVE dataset was adopted as the training set, while 10 scenes from the KAIST dataset were selected for testing. For the real data experiment, five real HSIs collected in TSA-Net [35] were used for evaluation. Each testing sample has 28 channels, with a spatial size of 660×660 pixels.

Implementation Details. We implemented Specformer on ubuntu20 using the PyTorch framework, and trained it using the Adam optimization algorithm on NVIDIA RTX3090. During training, the batchsize was set to 4, and the Specformer was trained for a total of 600 epochs with a learning rate of 0.0001 (400 epochs for state 1, 200 epochs for stage 2). When conducting simulation comparison, patches at a spatial size of 256×256 cropped from the 3D cubes were fed into the networks. As for real HSI reconstruction, the patch size was set to 660×660 to match the real-world measurements. The shifting step d in dispersion is set to 2 pixels. Consequently, the measurement size was 256×310 and 660×714 for simulation and real data experiment, respectively.

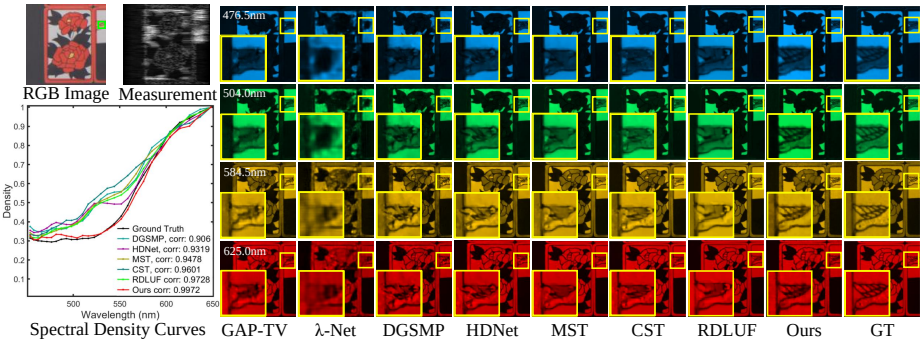

Fig. 4: Simulation reconstruction comparison of an exemplar Scene 7 with 4 out of 28 spectral channels. The results of 7 SOTA algorithms and our Specformer are presented. The spectral curves (bottom-left) correspond to the marked regions of the green box on the RGB image.

4.2 Quantitative Results

We compared the HSI reconstruction performance of our Specformer with other 11 SOTA methods, including 2 model-based methods (GAP-TV [34], and DeSCI

Table 1: Quantitative reconstruction comparison on 10 scenes. We adopt PSNR and SSIM [18] as the metrics to evaluate the HSI reconstruction performance. The best results were marked in bold.

Methods	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	Avg
GAP-TV [34]	26.82	22.89	26.31	30.65	23.64	21.85	23.76	21.98	22.63	23.10	24.36
	0.754	0.610	0.802	0.852	0.703	0.663	0.688	0.655	0.682	0.584	0.669
DeSCI [28]	27.13	23.04	26.62	34.96	23.94	22.38	24.45	22.03	24.56	23.59	25.27
	0.748	0.620	0.818	0.897	0.706	0.683	0.743	0.673	0.732	0.587	0.721
λ-net [38]	30.10	28.49	27.73	37.01	26.19	28.64	26.47	26.09	27.50	27.13	28.53
	0.849	0.805	0.870	0.934	0.817	0.853	0.806	0.831	0.826	0.816	0.841
TSA-Net [35]	32.03	31.00	32.25	39.19	29.39	31.44	30.32	29.35	30.01	29.59	31.46
	0.892	0.858	0.915	0.953	0.884	0.908	0.878	0.888	0.890	0.874	0.894
DGSMP [20]	33.26	32.09	33.06	40.54	28.86	33.08	30.74	31.55	31.66	31.44	32.63
	0.915	0.898	0.925	0.964	0.882	0.937	0.886	0.923	0.911	0.925	0.917
HDNet [19]	35.14	35.67	36.03	42.30	32.69	34.46	33.67	32.48	34.89	32.38	34.97
	0.935	0.940	0.943	0.969	0.946	0.952	0.926	0.941	0.942	0.937	0.943
MST [3]	35.40	35.87	36.51	42.27	32.77	34.80	33.66	32.67	35.39	32.50	35.18
	0.941	0.944	0.953	0.973	0.947	0.955	0.925	0.948	0.949	0.941	0.948
CST [2]	35.96	36.85	38.16	42.44	33.25	35.72	34.86	34.34	36.51	33.09	36.12
	0.949	0.955	0.962	0.975	0.955	0.963	0.944	0.961	0.957	0.945	0.957
DAUHST [5]	37.25	39.02	41.05	46.15	35.80	37.08	37.57	35.10	40.02	34.59	38.36
	0.958	0.967	0.971	0.983	0.969	0.970	0.963	0.966	0.970	0.956	0.967
PADUT [25]	37.36	40.43	42.38	46.62	36.26	37.27	37.83	35.33	40.86	34.55	38.89
	0.962	0.978	0.979	0.990	0.974	0.974	0.966	0.974	0.978	0.963	0.974
RDLUF [11]	37.94	40.95	43.25	**47.83**	37.11	37.47	38.58	35.50	41.83	35.23	39.57
	0.966	0.977	0.979	**0.990**	0.976	0.975	0.969	0.970	0.978	0.962	0.974
Specformer	**38.82**	**41.93**	**43.98**	47.77	**38.78**	**38.61**	**39.91**	**36.72**	**42.82**	**36.73**	**40.61**
	0.973	**0.982**	**0.983**	0.989	**0.983**	**0.982**	**0.977**	**0.982**	**0.985**	**0.969**	**0.981**

[28]), 3 CNN-based methods (λ-net [38], TSA-Net [35], and DGSMP [20]), and 6 recent Transformer-based methods (HDNet [19], MST [3], CST [2], DAUHST [5], PADUT [25] and RDLUF [11]). All the techniques were trained using the same datasets and evaluated under the same settings as DGSMP [20]. The quantitative reconstruction results on the 10 scenes of the KAIST dataset are presented in Tab. 1. Compared to DGSMP [20], MST [3], CST [2], DAUHST [5], and RDLUF [11], the reported Specformer method achieved PSNR improvements of 7.98 dB, 5.43dB, 4.49dB, 2.25dB and 1.04dB on average, respectively. Additionally, the reported method requires lower memory and computational costs as shown in Fig. 1. This demonstrates the effectiveness of simultaneously embedding the spatial sparsity and inter-spectral similarity of HSI into the learning process. It also validates that the parallel design can help the SF-block to model cross-window connections, and enlarge the receptive fields while maintaining linear complexity.

4.3 Qualitative Results

Simulation HSI Reconstruction. Figure 4 shows the visualized HSI reconstruction results of 7 SOTA methods and our Specformer technique. From the reconstructed HSIs and the zoom-in patches of the selected yellow boxes, we can see that the reconstruction performance of previous methods in texture-rich regions and edge regions is unsatisfactory. They either produce overly smooth results, sacrificing fine-grained structural content and textural detail, or introduce undesirable color artifacts and speckled textures. In contrast, our Specformer can accurately reconstruct the details of texture-rich regions and edge regions, as well as preserve the spatial smoothness of the homogeneous regions. This is because the uncertainty-driven self-adoptive loss can reinforce the network's attention on the regions with rich textures and edges, thus improving the HSI reconstruction quality of these regions. In addition, we plot the spectral density curves (bottom-left) corresponding to the picked region of the green box in the RGB image (top-left). The highest correlation and coincidence between our curve and the ground truth demonstrate the spectral-wise consistency restoration effectiveness of our Specformer. This is because FWSA can accurately model the spectral-wise long-distance dependencies and capture the inter-spectral similarities.

Real HSI Reconstruction. We further applied the approaches to real HSI reconstruction. Similar to DGSMP [20], we retrained all the networks on all scenes of CAVE [41] and KAIST [9] datasets. The reconstruction results on real measurements are presented in Fig. 5, from which we can see that compared with existing methods, our Specformer technique produced higher reconstruction quality in texture-rich and edge regions. Meanwhile, Specformer also produced better performance in suppressing noise.

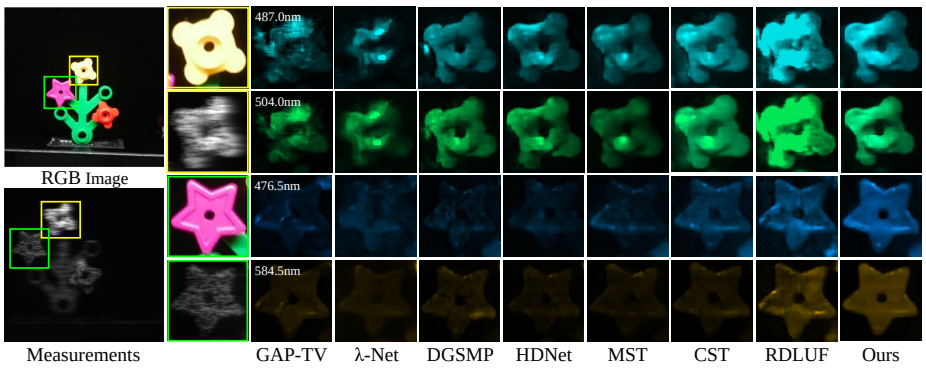

Fig. 5: Real reconstruction comparison of an exemplar Scene 1 with 4 out of 28 spectral channels. The results of 7 SOTA algorithms and our Specformer are presented. Please zoom in for a better view.

5 Ablation Study

To validate the effectiveness of each component in the Specformer network, we conducted a series of ablation studies on the CAVE [41] and KAIST [9] datasets. We consider several factors including the FWSA and LWSA modules in the SF-block, and the uncertainty-driven loss (UDL) function. The comparison results are presented in Tab. 2. The baseline (BL) model is derived by removing SF-block and UDL from Specformer. The "Serial" and "Parallel" represent combining the LWSA and FWSA modules using serial and parallel design, respectively.

Effectiveness of SF-Blcok. From Tab. 2, we can see that the absence of any component in the SF-block will result in performance degradation, which demonstrates the effectiveness of each component in the SF-block and the effectiveness of their combination. Moreover, the reconstruction performance of "BL+FWSA" is better than that of "BL+LWSA". This is because spectral representations are spatially sparse and spectrally highly self-similar. Hence, capturing spatial interactions may be less effective than modeling inter-spectra dependencies. However, only using FWSA cannot model the spatial sparsity, which is why the reconstruction performance of "BL+FWSA" is not as good as "BL+LWSA+FWSA". Furthermore, we can see that using a parallel design to combine FWSA and LWSA modules results in better reconstruction performance than using a serial design. Moreover, the computational and memory costs of parallel design (2.48M, 39.85G) are also lower than that of serial design (2.82M, 46.73G). This is because parallel design helps the SF-block to model cross-window connections, enlarges the receptive fields while maintaining linear complexity, and enables the complementary fusion of frequency-wise and spatial-wise features.

Table 2: Break-down ablation study. The models of different combinations were trained on the CAVE dataset and tested on the KAIST dataset. The first row is the baseline model.

BL	LWSA	FWSA	Serial	Parallel	UDL	PSNR	SSIM	Params	GFLOPs
✓						31.08	0.874	1.42M	14.87
✓	✓					33.34	0.927	1.89M	23.14
✓		✓				35.97	0.958	1.67M	19.79
✓	✓	✓	✓			37.58	0.966	2.82M	46.73
✓	✓	✓		✓		39.82	0.978	2.48M	39.85
✓	✓	✓		✓	✓	**40.61**	**0.981**	2.48M	39.85

Effectiveness of UDL. We further investigated the contribution of the UDL function. The visualization results in Fig. 6 demonstrate the UDL's ability to enhance the reconstruction quality in texture-rich and edge regions. From Tab. 2, we can also see that by enhancing the HSI reconstruction quality for the regions with rich textures and edges, the PSNR and SSIM of Specformer's

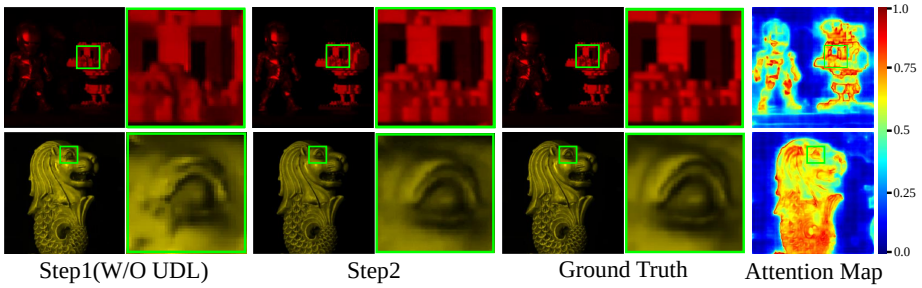

Fig. 6: Ablation study of the UDL. Step1 represents the HSI reconstructed by the Specformer that has not been trained by Loss_{UDL}, and step2 represents the HSI reconstructed by the specformer that has been trained by Loss_{UDL}.

reconstruction results are increased. Moreover, we can see from Tab. 2 that the improvements achieved by UDL do not bring any additional memory (Params) and computational (GFLops) costs during testing. This is because the two-stage training strategy of UDL does not change the structure of the final model, but only increases the training time by about 30%. However, owing to the low memory and computational costs of Specformer, we can still complete training within 6 hours on a single RTX 3090 GPU.

6 Conclusion

In this work, we explored how to simultaneously embed the spatial sparsity and inter-spectral similarity nature of HSI into the learning-based reconstruction process. To this end, we proposed a novel uncertainty-driven method, termed Specformer, for HSI reconstruction. Specifically, we first introduced FWSA module, and used a parallel design to combine it with an LWSA module to form an SF-block. LWSA can guide the network to focus on the image regions with dense spectral information. FWSA can model the inter-spectral similarity. Parallel design helps the SF-block to model cross-window connections, and enlarge its receptive fields while maintaining linear complexity. We inserted the SF-block as the main building block in a U-shape encoder-decoder architecture to form Specformer. In addition, considering that texture-rich and edge regions in HSI are more difficult to reconstruct than smooth regions, we designed an uncertainty-driven self-adaptive loss function, which assigns greater priority to the regions with rich textures and edges during training, thus improving the reconstruction quality of these regions. Extensive quantitative and qualitative experiments demonstrate that the reported Specformer technique outperforms other SOTA methods with lower computational and memory costs.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Nos. 62322502, 61827901, 62131003), the Guangdong Province Key Laboratory of Intelligent Detection in Complex Environment of Aerospace, Land and Sea (2022KSYS016), and the Guangdong Cross domain Intelligent Detection and Information Processing Innovation Team (2023KCXT044).

References

1. Borengasser, M., Hungate, W.S., Watkins, R.: Hyperspectral remote sensing: principles and applications. CRC press (2007)
2. Cai, Y., Lin, J., Hu, X., Wang, H., Yuan, X., Zhang, Y., Timofte, R., Gool, L.V.: Coarse-to-fine sparse transformer for hyperspectral image reconstruction. In: ECCV (2022)
3. Cai, Y., Lin, J., Hu, X., Wang, H., Yuan, X., Zhang, Y., Timofte, R., Gool, L.V.: Mask-guided spectral-wise transformer for efficient hyperspectral image reconstruction. In: CVPR (2022)
4. Cai, Y., Lin, J., Wang, H., Yuan, X., Ding, H., Zhang, Y., Timofte, R., Gool, L.V.: Degradation-aware unfolding half-shuffle transformer for spectral compressive imaging. NIPS **35**, 37749–37761 (2022)
5. Cai, Y., Lin, J., Wang, H., Yuan, X., Ding, H., Zhang, Y., Timofte, R., Gool, L.V.: Degradation-aware unfolding half-shuffle transformer for spectral compressive imaging. In: NIPS (2022)
6. Cao, X., Yue, T., Lin, X., Lin, S., Yuan, X., Dai, Q., Carin, L., Brady, D.J.: Computational snapshot multispectral cameras: Toward dynamic capture of the spectral world. IEEE Signal Process. Mag. **33**(5), 95–108 (2016)
7. Chan, S.H., Wang, X., Elgendy, O.A.: Plug-and-play admm for image restoration: Fixed-point convergence and applications. IEEE T COMPUT IMAG **3**(1), 84–98 (2016)
8. Chang, J., Lan, Z., Cheng, C., Wei, Y.: Data uncertainty learning in face recognition. In: CVPR. pp. 5710–5719 (2020)
9. Choi, I., Kim, M., Gutierrez, D., Jeon, D., Nam, G.: High-quality hyperspectral reconstruction using a spectral prior. Tech. rep. (2017)
10. Der Kiureghian, A., Ditlevsen, O.: Aleatory or epistemic? does it matter? Structural safety **31**(2), 105–112 (2009)
11. Dong, Y., Gao, D., Qiu, T., Li, Y., Yang, M., Shi, G.: Residual degradation learning unfolding framework with mixing priors across spectral and spatial for compressive spectral imaging. In: CVPR. pp. 22262–22271 (2023)
12. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
13. Du, H., Tong, X., Cao, X., Lin, S.: A prism-based system for multispectral video acquisition. In: ICCV. pp. 175–182. IEEE (2009)
14. Figueiredo, M.: Adaptive sparseness using jeffreys prior. Advances in neural information processing systems **14** (2001)

15. Figueiredo, M.A., Nowak, R.D., Wright, S.J.: Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. IEEE J-STSP **1**(4), 586–597 (2007)
16. Fu, Y., Zhang, T., Wang, L., Huang, H.: Coded hyperspectral image reconstruction using deep external and internal learning. IEEE TPAMI **44**(7), 3404–3420 (2021)
17. Gu, Y., Jin, Z., Chiu, S.C.: Active learning combining uncertainty and diversity for multi-class image classification. IET Computer Vision **9**(3), 400–407 (2015)
18. Horé, A., Ziou, D.: Image quality metrics: Psnr vs. ssim. In: ICPR. pp. 2366–2369 (2010). https://doi.org/10.1109/ICPR.2010.579
19. Hu, X., Cai, Y., Lin, J., Wang, H., Yuan, X., Zhang, Y., Timofte, R., Gool, L.V.: Hdnet: High-resolution dual-domain learning for spectral compressive imaging. In: CVPR (2022)
20. Huang, T., Dong, W., Yuan, X., Wu, J., Shi, G.: Deep gaussian scale mixture prior for spectral compressive imaging. In: CVPR. pp. 16216–16225 (2021)
21. Kendall, A., Gal, Y.: What uncertainties do we need in bayesian deep learning for computer vision? NIPS **30** (2017)
22. Kim, M.H., Harvey, T.A., Kittle, D.S., Rushmeier, H., Dorsey, J., Prum, R.O., Brady, D.J.: 3d imaging spectroscopy for measuring hyperspectral patterns on solid objects. TOG **31**(4), 1–11 (2012)
23. Kittle, D., Choi, K., Wagadarikar, A., Brady, D.J.: Multiframe image estimation for coded aperture snapshot spectral imagers. Applied optics **49**(36), 6824–6833 (2010)
24. Lee, C., Chung, K.S.: Gram: Gradient rescaling attention model for data uncertainty estimation in single image super resolution. In: ICMLA. pp. 8–13. IEEE (2019)
25. Li, M., fu, Y., Liu, J., Zhang, Y.: Pixel adaptive deep unfolding transformer for hyperspectral image reconstruction. In: ICCV (2023)
26. Li, S., Xue, K., Zhu, B., Ding, C., Gao, X., Wei, D., Wan, T.: Falcon: A fourier transform based approach for fast and secure convolutional neural network predictions. In: CVPR. pp. 8705–8714 (2020)
27. Lin, X., Liu, Y., Wu, J., Dai, Q.: Spatial-spectral encoded compressive hyperspectral imaging. ACM Transactions on Graphics (TOG) **33**(6), 1–11 (2014)
28. Liu, Y., Yuan, X., Suo, J., Brady, D.J., Dai, Q.: Rank minimization for snapshot compressive imaging. IEEE TPAMI **41**(12), 2990–3006 (2018)
29. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: ICCV. pp. 10012–10022 (2021)
30. Llull, P., Liao, X., Yuan, X., Yang, J., Kittle, D., Carin, L., Sapiro, G., Brady, D.J.: Coded aperture compressive temporal imaging. Optics express **21**(9), 10526–10545 (2013)
31. Lu, G., Fei, B.: Medical hyperspectral imaging: a review. J BIOMED OPT **19**(1), 010901–010901 (2014)
32. Ma, J., Liu, X.Y., Shou, Z., Yuan, X.: Deep tensor admm-net for snapshot compressive imaging. In: ICCV. pp. 10223–10232 (2019)
33. Melgani, F., Bruzzone, L.: Classification of hyperspectral remote sensing images with support vector machines. IEEE Trans Geosci Remote Sens **42**(8), 1778–1790 (2004)
34. Meng, Z., Jalali, S., Yuan, X.: Gap-net for snapshot compressive imaging. arXiv preprint arXiv:2012.08364 (2020)
35. Meng, Z., Ma, J., Yuan, X.: End-to-end low cost compressive spectral imaging with spatial-spectral self-attention. In: ECCV. pp. 187–204. Springer (2020)

36. Meng, Z., Qiao, M., Ma, J., Yu, Z., Xu, K., Yuan, X.: Snapshot multispectral endomicroscopy. Optics Letters **45**(14), 3897–3900 (2020)
37. Meng, Z., Yu, Z., Xu, K., Yuan, X.: Self-supervised neural networks for spectral snapshot compressive imaging. In: ICCV. pp. 2622–2631 (2021)
38. Miao, X., Yuan, X., Pu, Y., Athitsos, V.: l-net: Reconstruct hyperspectral images from a snapshot measurement. In: ICCV. pp. 4059–4069 (2019)
39. Nussbaumer, H.J., Nussbaumer, H.J.: The fast Fourier transform. Springer (1982)
40. Pan, Z., Healey, G., Prasad, M., Tromberg, B.: Face recognition in hyperspectral images. IEEE TPAMI **25**(12), 1552–1560 (2003)
41. Park, J.I., Lee, M.H., Grossberg, M.D., Nayar, S.K.: Multispectral imaging using multiplexed illumination. In: ICCV. pp. 1–8. IEEE (2007)
42. Pitas, I.: Digital image processing algorithms and applications. John Wiley & Sons (2000)
43. Qiao, M., Liu, X., Yuan, X.: Snapshot spatial–temporal compressive imaging. Optics letters **45**(7), 1659–1662 (2020)
44. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: MICCAI. pp. 234–241. Springer (2015)
45. Vaibhav, V.: Fast inverse nonlinear fourier transform. Physical Review E **98**(1), 013304 (2018)
46. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. NIPS **30** (2017)
47. Wagadarikar, A., John, R., Willett, R., Brady, D.: Single disperser design for coded aperture snapshot spectral imaging. Applied optics **47**(10), B44–B51 (2008)
48. Wagadarikar, A.A., Pitsianis, N.P., Sun, X., Brady, D.J.: Video rate spectral imaging using a coded aperture snapshot spectral imager. Optics express **17**(8), 6368–6388 (2009)
49. Wang, L., Xiong, Z., Gao, D., Shi, G., Wu, F.: Dual-camera design for coded aperture snapshot spectral imaging. Applied optics **54**(4), 848–858 (2015)
50. Wang, L., Xiong, Z., Shi, G., Wu, F., Zeng, W.: Adaptive nonlocal sparse representation for dual-camera compressive hyperspectral imaging. IEEE TPAMI **39**(10), 2104–2111 (2016)
51. Yuan, X.: Generalized alternating projection based total variation minimization for compressive sensing. In: ICIP. pp. 2539–2543. IEEE (2016)
52. Yuan, X., Brady, D.J., Katsaggelos, A.K.: Snapshot compressive imaging: Theory, algorithms, and applications. IEEE Signal Process. Mag. **38**(2), 65–88 (2021)
53. Zhang, S., Wang, L., Fu, Y., Zhong, X., Huang, H.: Computational hyperspectral imaging based on dimension-discriminative low-rank tensor recovery. In: ICCV. pp. 10183–10192 (2019)