Latent Diffusion Prior Enhanced Deep Unfolding for Snapshot Spectral Compressive Imaging

Zongliang Wu[1,2*], Ruiying Lu[3*], Ying Fu[4], and Xin Yuan[2(✉)]

[1] Zhejiang University, Hangzhou, China
[2] School of Engineering, Westlake University, Hangzhou, China.
{wuzongliang,xyuan}@westlake.edu.cn
[3] School of Cyber Engineering, Xidian University, Xi'an, China.
ruiyinglu_xidian@163.com
[4] School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China. fuying@bit.edu.cn

Abstract. Snapshot compressive spectral imaging reconstruction aims to reconstruct three-dimensional spatial-spectral images from a single-shot two-dimensional compressed measurement. Existing state-of-the-art methods are mostly based on deep unfolding structures but have intrinsic performance bottlenecks: *i*) the ill-posed problem of dealing with heavily degraded measurement, and *ii*) the regression loss-based reconstruction models being prone to recover images with few details. In this paper, we introduce a generative model, namely the latent diffusion model (LDM), to generate degradation-free prior to enhance the regression-based deep unfolding method by a two-stage training procedure. Furthermore, we propose a Trident Transformer (TT), which extracts correlations among prior knowledge, spatial, and spectral features, to integrate knowledge priors in deep unfolding denoiser, and guide the reconstruction for compensating high-quality spectral signal details. To our knowledge, this is the first approach to integrate physics-driven deep unfolding with generative LDM in the context of CASSI reconstruction. Comparisons on synthetic and real-world datasets illustrate the superiority of our proposed method in both reconstruction quality and computational efficiency. The code is available at https://github.com/Zongliang-Wu/LADE-DUN.

Keywords: Spectral imaging · Deep unfolding · Diffusion model

1 Introduction

In contrast to normal RGB images which only have three spectral bands, hyperspectral images (HSIs) contain multiple spectral bands with more diverse spectral information. The spectral information serves to characterize distinct objects assisting high-level image tasks [31, 35, 50, 59, 60] and the observation of the world like medical imaging [38, 51] and remote sensing [18, 39]. However, the

*Z. Wu and R. Lu—Contribute equally

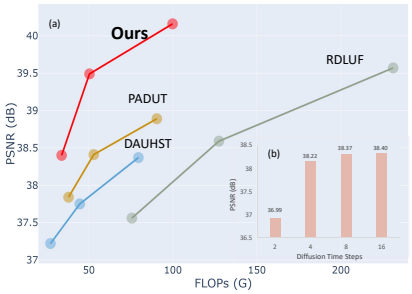

Fig. 1: (a) Comparison of PSNR (dB)-FLOPs (G) with previous HSI reconstruction methods. (b) The ablation study of using different time steps in diffusion. Our method achieves the desired results by only very few steps.

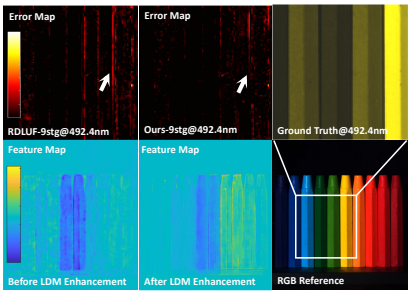

Fig. 2: The top row: the error maps of the previous SOTA and our method. The bottom row: the feature map before and after applying LDM enhancement. The enhanced features demonstrate less noise and clearer edges.

capture of HSIs is a question that has been studied for a long time because we need to collect 3-dimensional (3D) HSI signals by 2D sensors.

For many years, scientists have focused on how to collect HSIs in a quick and convenient method. In 2007, based on compressive sensing theory, a single-shot compressive spectral imaging way [17] was created to efficiently collect HSIs, named coded aperture snapshot spectral imaging (CASSI). The later improvement works [44, 61] provide better imaging quality and lower cost. CASSI modulates the HSI signal across various bands and combine all the modulated spectra to produce a 2D compressed measurement. Consequently, the task of reconstructing the 3D HSI signals from the 2D compressive measurements presents a fundamental challenge for the CASSI system.

The reconstruction process can be viewed as solving an ill-posed problem. Many attempts at solving this problem including traditional model-based methods [1, 2, 70] and the learning-based methods [9, 45, 47] have been proposed since the inception of CASSI system. The deep unfolding network (DUN) is a combination of convex optimization and neural network prior (denoiser), enjoying both the interpretability of the model-based method and the power of learning-based methods. This branch of methods leads the development trend in recent years [6, 15, 32, 46, 64] and achieves SOTA performance.

However, unlike super-resolution or deblurring that recovers from natural images, CASSI reconstruction has to recover HSIs from the compressed domain measurements, which results in severe degradation according to physical modulation, spectral compression, and multiple types of system noises. Thus, the CASSI reconstruction problem is much harder to learn intrinsic HSI properties than the normal image restoration tasks [30, 33, 34, 40, 66, 75, 76]. In the unfolding framework of the CASSI reconstruction method, the denoising network plays a critical role in deciding the final performance, which is embedded in each stage of the DUN. However, it always suffers from the performance bottleneck due to

the intrinsic ill-posed problem of dealing with heavily degraded measurements. Thus, a high-performance denoiser with degradation-free knowledge is desired for CASSI reconstruction. Another problem is that previous popular regression-based reconstruction methods have difficulty in recovering details because the widely used regression losses are conservative with high-frequency details [55].

To address these challenges, we introduce a generative prior in this paper to guide the reconstruction process in an unfolding framework. During training, the prior will be first learned from clean HSIs by an image encoder and then generated by a Latent Diffusion Model (LDM) from Gaussian noise and compressed measurement. Then, the learned prior is embedded into the deep denoiser of the DUN by a prior-guided Transformer. Significantly, our DUN is able to leverage external prior knowledge from clean HSIs and the powerful generative ability of LDM enhancing its reconstruction performance. The primary contributions presented in this paper can be summarized as follows:

i) We propose a novel **LDM-based unfolding network** for CASSI reconstruction, where the clean image priors are generated by a latent diffusion model to facilitate high-quality hyperspectral reconstruction. There is no additional data or training time required. To the best of our knowledge, this is the first attempt to combine the physics-driven deep unfolding with generative LDM in CASSI reconstruction.

ii) We design a three-in-one Transformer structure dubbed Trident Transformer (TT) to extract the correlation among prior knowledge, spatial, and spectral features. In TT, motivated by pansharpening techniques, we introduce an asymmetric cross-scale multi-head self-attention (ACS-MHSA) mechanism designed to efficiently fuse spatial-spectral features.

iii) Extensive experiments on the synthetic benchmark and real dataset demonstrate the superior quantitative performances (Fig. 1), visual quality (Fig. 2), and lower computational cost of our proposed method.

2 Related Work

2.1 Diffusion Model in Low-level Vision

Diffusion models (DMs) [22,56] are probabilistic generative models, which model the data distribution by learning a gradual iterative denoising process from the Gaussian distribution to the data distribution. Notably, they demonstrate promising capabilities in generating high-quality samples that encompass a wide range of modes, including super-resolution [16] and inpainting [41]. In light of the impressive achievements of diffusion models in image domains, numerous research endeavors [3, 19, 21, 23] have extended it to video generation. However, diffusion models suffer from significant computation inefficiency regarding data sampling, primarily due to the iterative denoising process required for inference. To address this challenge, several methods propose effective sampling techniques from trained diffusion models [57, 73], or alternatively learning the

data distribution from a low-dimensional latent space [53], *i.e.* the latent diffusion model. The latent diffusion has a relatively faster speed and powerful generative ability for super-resolution and inpainting, but similar to the normal diffusion model, it is also prone to issues such as misaligned distribution of fine details and the occurrence of unwanted artifacts, leading to suboptimal performance in distortion-based metrics, e.g., PSNR. Moreover, latent diffusion costs large computational resources both for training and inference due to its large-size encoder and denoiser. Towards this end, some works combine the generative diffusion model with the regression restoration network and work well on distortion-based metrics like deblurring [52]. The recent works [12, 68] employ LDM on many low-level vision tasks and achieve SOTA with reasonable computational cost. We name these methods 'integrated diffusion' to distinguish them from the pure diffusion method. Nevertheless, employing diffusion models for the efficient reconstruction of hyperspectral images from highly compressed and degraded measurements presents significant challenges.

2.2 Hyperspectral Image Reconstruction

Before the advent of the deep learning wave, traditional model-based methods iteratively solved this inverse problem by convex optimization [62, 63, 74] with some hand-crafted constraints based on image priors, like sparsity [29] and low-rank [37]. These methods are robust and interpretable but require manual parameter tuning with low reconstruction speed and performance. With the help of deep learning, Plug-and-play (PnP) algorithms [8, 10, 11, 49, 54, 71, 72], embeds pre-trained denoising networks into convex optimization to solve the reconstruction problem, but still has limitations on performance because of the pre-trained denoiser. In recent years, the End-to-end (E2E) reconstruction directly trains a powerful deep neural network, like convolutional neural network (CNN) [13, 25, 39] and Transformers [4, 5, 7], to learn the recovery process from inputs (measurements) to outputs (desired HSIs). However, this simple design lacks interpretability and robustness for various hardware systems. Therefore, an interpretable design of a reconstruction network that unfolds a convex optimization process named DUN is proposed to leverage these problems. A series of CASSI reconstruction works based on DUN [6, 15, 32, 42, 43, 69] are proposed and become the state-of-the-art (SOTA) method. DUN can combine both interpretability in model-based methods and performance in deep learning-based methods to reconstruct CASSI at a fast speed. It changes iterative steps in optimization into several stages in a single network. The prior for optimization becomes a deep neural network denoiser. Since the DUN needs to define the forward model of imaging, it is also considered a physics-driven network. However, the recent DUNs still have bottlenecks for their regression-based denoiser design and the difficulty of dealing with compressive measurement features. Bearing these concerns, we propose an 'integrated diffusion' module and integrate it into the physics-driven DUN framework and design an efficient way to aggregate complex features during reconstruction.

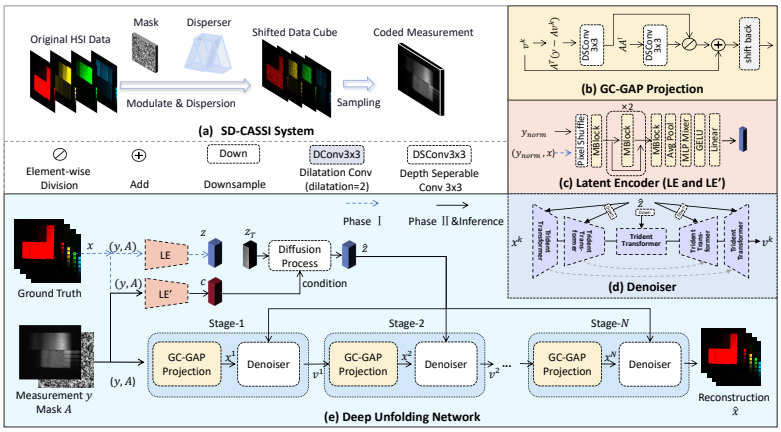

Fig. 3: (a) The single disperser CASSI imaging process. HSI data cube is captured by a monochromatic sensor. (b) GC-GAP projection. (c) Latent encoder. (d) Simplified Denoiser. (e) The measurement $\boldsymbol{y}$ and masks $\boldsymbol{A}$ pass through an N-stage DUN, where each stage is composed of a GC-GAP projection and a denoiser. The denoiser follows a U-shape structure and consists of five Trident Transformers (TT), where each TT is assisted with prior knowledge $\boldsymbol{z}_{GT}$ generated from the diffusion model.

3 Problem Formulation

The CASSI system has high efficiency in capturing 3D spectral signals by initially coding spectral data with different wavelengths in an aperture and then integrating them into a 2D monochromatic sensor. The mathematical forward process of the widely used single-disperser CASSI (SD-CASSI) [62] can be illustrated as Fig. 3 (a). As can be seen, the original HSI data, denoted as $\boldsymbol{X} \in \mathbb{R}^{W\times H\times N_\lambda}$, is coded by the physical mask $\boldsymbol{M} \in \mathbb{R}^{W\times H}$, where W, H, and N_λ denote the width, height, and the number of spectral channels, respectively. The coded HSI data cube is represented as $\mathbf{X}'(:,:,n_\lambda) = \mathbf{X}(:,:,n_\lambda) \odot \mathbf{M}, n_\lambda = 1, 2, \ldots, N_\lambda$, where $\odot$ represents the element-wise multiplication. After light propagating through the disperser, each channel of $\mathbf{X}'$ is shifted along the $\boldsymbol{H}$-$axis$. The shifted data cube is denoted as $\mathbf{X}'' \in \mathbb{R}^{W\times \tilde{H}\times N_\lambda}$, where $\tilde{H} = H + d_\lambda$. d_λ is the shifted distance of the N_λ-th wavelength. This process can be formulated as modulating the shifted version $\tilde{\mathbf{X}} \in \mathbb{R}^{W\times \tilde{H}\times N_\lambda}$ with a shifted mask $\tilde{\mathbf{M}} \in \mathbb{R}^{W\times \tilde{H}\times N_\lambda}$, where $\tilde{\boldsymbol{M}}(i,j,n_\lambda) = \boldsymbol{M}(w, h + d_\lambda)$. At last, the imaging sensor captures the shifted image into a 2D measurement $\mathbf{Y}$, calculated as $\mathbf{Y} = \sum_{n_\lambda=1}^{N_\lambda} \tilde{\mathbf{X}}(:,:,n_\lambda) \odot \tilde{\mathbf{M}}(:,:,n_\lambda) + \mathbf{B}$, where $\mathbf{B} \in \mathbb{R}^{W\times \tilde{H}}$ denotes the measurement noise. By vectorizing the data cube and measurement, that is $\boldsymbol{x} = \texttt{vec}(\tilde{\mathbf{X}}) \in \mathbb{R}^{W\tilde{H}N_\lambda}$ and $\boldsymbol{y} = \texttt{vec}(\mathbf{Y}) \in \mathbb{R}^{W\tilde{H}}$, this model can be formulated as

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}, \tag{1}$$

where $\boldsymbol{A} \in \mathbb{R}^{W\tilde{H}\times W\tilde{H}N_\lambda}$ denotes the sensing matrix (coded aperture) which is a concatenation of diagonal matrices, that is $\boldsymbol{A} = [\mathbf{D}_1, \ldots, \mathbf{D}_\lambda]$, where $\mathbf{D}_\lambda =$

$Diag(vec(\tilde{\mathbf{M}}(:,:,n_\lambda)))$ is the diagonal matrix with $vec(\tilde{\mathbf{M}}(:,:,n_\lambda))$ as the diagonal elements. In this paper, we will propose a method to solve the ill-posed problem, reconstructing the HSI $\boldsymbol{x}$ from the compressed measurement $\boldsymbol{y}$.

4 Proposed Model

To solve the problem in Eq. (1), we proposed a novel unfolding enhanced by latent diffusion prior. As shown in Fig. 3(e), in the inference phase, the measurement and masks pass through an N-stage DUN, where each stage is composed of a GC-GAP projection and a denoiser. The denoiser follows a U-shape structure and consists of Trident Transformers (Fig. 3(d)), where each TT is assisted with the LDM prior. We'll describe these modules in more detail in this section.

4.1 The Unfolding GAP Framework

Eq. (1) can be typically solved by convex optimization by the objective below:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \tfrac{1}{2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \tau R(\boldsymbol{x}), \tag{2}$$

where τ is a noise-balancing factor. The first term guarantees that the solution $\hat{\boldsymbol{x}}$ fits the measurement, and the term $R(\boldsymbol{x})$ refers to the image regularization.

To solve the optimization problem, we employ GAP (Generalized Alternating Projection) as our optimization framework, which extends classical alternating projection to the case in which projections are performed between convex sets that undergo a systematic sequence of changes. It can be interrupted anytime to return a valid solution and resumed subsequently to improve the solution [36]. This property is very suitable for DUN which has very limited 'optimization iterations' (stages in the DUN). Specifically, we introduce an auxiliary parameter $\boldsymbol{v}$, Eq. (2) can be written as:

$$(\hat{\boldsymbol{x}}, \hat{\boldsymbol{v}}) = \arg\min_{\boldsymbol{x},\boldsymbol{v}} \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{v}\|_2^2 + \tau R(\boldsymbol{v}), \quad \text{s.t.} \quad \boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}. \tag{3}$$

Then, the problem can be solved by the following sub-problems: Firstly, we aim at updating $\boldsymbol{x}$:

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{v}^{(k)} + \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{v}^{(k)}). \tag{4}$$

This step projects measurement to a 3D signal space by Euclidean projection. Secondly, we aim at updating v:

$$\boldsymbol{v}^{(k+1)} = \mathcal{D}_{k+1}(\boldsymbol{x}^{(k+1)}, \boldsymbol{z}), \tag{5}$$

where $\mathcal{D}_k$ is the neural network denoiser of the $k-th$ stage and $\boldsymbol{z}$ is prior knowledge which will be described in Sec. 4.2. This step tries to map $\boldsymbol{x}^{(k+1)}$ to

the target signal domain. Considering the projection step Eq. (4), assisted by deep network, we can modify it as follows:

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{v}^{(k)} + \mathrm{DSC}(\boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{v}^{(k)})), \tag{6}$$

where DSC(·) denotes a set of depthwise separable convolution and GELU [20] operations. The detailed process is shown in Fig. 3(b). Considering the stage number is much less than the iteration numbers in traditional model-based methods, it is difficult to achieve convergence with limited steps of gradient descent. Thus, we utilize these learnable parameters to rectify the gradients in the limited stage, and we refer to this method as the Gradient Correction GAP (GC-GAP). The overall unfolding framework is shown in Fig. 3(e), where mask A and measurement y are inputs of the network. According to the Eq. (6) and (5), the first stage outputs $\boldsymbol{v}^1$ can be obtained.

4.2 Latent Diffusion Prior Assisted Unfolding Denoising

The denoising process in DUN leads to a natural performance bottleneck due to the intrinsic problem of heavily degraded input. Thus, we introduce external degradation-free prior knowledge to compensate for the denoising process. We will then introduce this process in a two-phase manner.

Phase I: Learning Prior Knowledge from clean HSIs. In this phase, we use an image encoder to compress both compressive measurement $\boldsymbol{y}$ and clean HSIs (Ground-Truth hyperspectral images) $\boldsymbol{x}$ into latent space. However, instead of simply using measurement $\boldsymbol{y}$, we transfer $\boldsymbol{y}$ by Euclidean projection to 3D HSIs space and normalize it by sensing matrix $\boldsymbol{y}_{norm} \in \mathbb{R}^{W\times H\times N_\lambda} = \boldsymbol{A}^\top\left(\boldsymbol{A}\boldsymbol{A}^\top\right)^{-1}\boldsymbol{y}$. This will improve the balance between two different inputs and easier for the encoder to learn their relation. The input of the encoder in the first phase is $\boldsymbol{I}_{\mathrm{E}} \in \mathbb{R}^{W\times H\times 2N_\lambda} = \mathrm{concatenate}(\boldsymbol{y}_{norm}, \boldsymbol{x})$. Thus the latent encoder process can be written as $\boldsymbol{z}_{GT} \in \mathbb{R}^{N\times C} = \mathrm{LE}(\boldsymbol{I}_{\mathrm{E}})$, where $N \ll W \times H$, C is the latent feature channel number. The LE can be seen in Fig. 3 (c): to alleviate the computational burden within LDM, we employ mobile blocks (MBlocks) [24], devoid of batch normalization instead of normal convolution. It aims at efficiently extracting representative visual features while maintaining computational efficiency. Moreover, considering the limitations in convolution, we add an MLP-Mixer [58] in LE to provide fast information exchange between patches by token-mixing MLP. Then the $\boldsymbol{z}_{GT}$ will be used as prior in the denoiser to compensate for the denoising errors. The DUN will reconstruct HSI signals using measurement and mask with the assistance of $\boldsymbol{z}_{GT}$, *i.e.* $\hat{\boldsymbol{x}} = \mathrm{DUN}(\boldsymbol{y}, \boldsymbol{A}, \boldsymbol{z}_{GT})$. Note that unlike the original LDM having its entire 'Auto Encoder', there is no corresponding 'decoder' specified here, because $\boldsymbol{z}$ is sent to DUN for 'decoding'. In this phase, we only use the reconstruction loss: $\mathcal{L}_{\mathrm{rec}} = \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_1$.

Phase II: Generating Prior by Latent Diffusion Model. After learning the prior representation from clean HSIs, we aim to learn an LDM to generate this prior condition on measurement $\boldsymbol{y}$ in the second phase. Specifically, the encoder in the first phase LE is fixed to encode clean HSIs and measurements

to $\boldsymbol{z}_{GT}$ as the generative object of latent space, *i.e.* the starting point of the forward Markov process in the diffusion model. Then as usual forward process, Gaussian noise will be gradually added on $\boldsymbol{z}_{GT}$ across T time steps according to the parameter β_t:

$$\begin{aligned} q\left(\boldsymbol{z}_{1:T} \mid \boldsymbol{z}_0\right) &= \textstyle\prod_{t=1}^{T} q\left(\boldsymbol{z}_t \mid \boldsymbol{z}_{t-1}\right), \forall t = 1, \ldots, T, \\ q\left(\boldsymbol{z}_t \mid \boldsymbol{z}_{t-1}\right) &= \mathcal{N}\left(\boldsymbol{z}_t; \sqrt{1-\beta_t}\boldsymbol{z}_{t-1}, \beta_t \mathbf{I}\right), \end{aligned} \tag{7}$$

where $\boldsymbol{z}_t$ represents the noisy features at the t-th step, and $\boldsymbol{z}_0 = \boldsymbol{z}_{GT}$ is the generative target. $\beta_{1:T} \in (0, 1)$ are hyperparameters that control the variance of the Gaussian distribution $\mathcal{N}$. Through iterative derivation with reparameterization [28], Eq. (7) can be written as:

$$\begin{aligned} q\left(\boldsymbol{z}_t \mid \boldsymbol{z}_0\right) &= \mathcal{N}\left(\boldsymbol{z}_t; \sqrt{\bar{\alpha}_t}\boldsymbol{z}_0, \left(1-\bar{\alpha}_t\right)\mathbf{I}\right), \\ \alpha = 1 - \beta_t, &\quad \bar{\alpha}_t = \textstyle\prod_{i=1}^{t} \alpha_i. \end{aligned} \tag{8}$$

The reverse process involves generating the prior features from a pure Gaussian distribution step-by-step condition on the measurement. The reverse process operates as a T-step Markov chain that runs backward from $\boldsymbol{z}_T$ to $\boldsymbol{z}_0$. Specifically, the posterior distribution of the reverse step from $\boldsymbol{z}_t$ to $\boldsymbol{z}_{t-1}$ can be formulated as:

$$\begin{aligned} q\left(\boldsymbol{z}_{t-1} \mid \boldsymbol{z}_t, \boldsymbol{z}_0\right) &= \mathcal{N}\left(\boldsymbol{z}_{t-1}; \boldsymbol{\mu}_t\left(\boldsymbol{z}_t, \boldsymbol{z}_0\right), \tfrac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t \mathbf{I}\right), \\ \boldsymbol{\mu}_t\left(\boldsymbol{z}_t, \boldsymbol{z}_0\right) &= \tfrac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{z}_t - \tfrac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}\right), \end{aligned} \tag{9}$$

where $\boldsymbol{\epsilon}$ represents the noise added on $\boldsymbol{z}_t$. Thus, a denoising network $\boldsymbol{\epsilon}_\theta$ is used to predict the noise $\boldsymbol{\epsilon}$ at each step, following the previous works [22, 53]. In order to encode condition $\boldsymbol{y}$ to latent space, another encoder LE' is applied to extract features, with the same structure as the LE of Phase I. Specifically, LE' compresses the normalized measurement $\boldsymbol{y}_{norm}$ into latent space to get the latent condition features $\boldsymbol{c} \in \mathbb{R}^{N \times C}$. In the end, we use the denoising network to predict the noise ϵ_t according to $\boldsymbol{z}_t$ of the previous step in reverse process and the condition $\boldsymbol{c}$, stated as $\boldsymbol{\epsilon} = \boldsymbol{\epsilon}_\theta\left(\boldsymbol{z}_t, \boldsymbol{c}, t\right)$. With the substitution of $\boldsymbol{\epsilon}_\theta$ in Eq. (9) and set the variance as $1 - \alpha_t$, the reverse inference can be stated as:

$$\boldsymbol{z}_{t-1} = \tfrac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{z}_t - \tfrac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta\left(\boldsymbol{z}_t, \boldsymbol{c}, t\right)\right) + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_t, \tag{10}$$

where $\boldsymbol{\epsilon}_t \sim \mathcal{N}(0, \mathbf{I})$. Finally, we can generate the target prior feature $\hat{\boldsymbol{z}} \in \mathbb{R}^{N \times C}$ after T iterative sampling $\boldsymbol{z}_t$ by Eq. (10). As shown in Fig. 3(e), the predicted prior feature is then used to guide the Transformer in denoiser. Notably, since the distribution of the latent space with the size of $\mathbb{R}^{N \times C}$ (e.g, 16×256) is much simpler than that of images with size $\mathbb{R}^{H \times W \times N_\lambda}$, the prior feature can be generated with a small number of iterations T, corresponding to paper [53].

Typically, training the diffusion model refers to training the denoising network $\boldsymbol{\epsilon}_\theta$. Following the previous works [22, 57], we train the model by optimizing

the weighted variational bound. The training objective is:

$$\nabla_{\boldsymbol{\theta}} \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta \left(\sqrt{\bar{\alpha}_t} \boldsymbol{z}_{GT} + \sqrt{1-\bar{\alpha}_t} \boldsymbol{\epsilon}, \boldsymbol{c}, t \right) \right\|_2^2, \tag{11}$$

where $\boldsymbol{z}_{GT}$ and $\boldsymbol{c}$ are ground-truth prior features and the latent condition representations defined above; $t \in [1, T]$ is the randomly sampled time step; $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$ denotes the sampled Gaussian noise. We employ DDPM [22] for diffusion as the original LDM [53]. Considering small prior size and time efficiency, we adopt a simple denoising network consisting of several MLP layers for fast diffusion denoising. All the parameters are jointly updated in the network with the objective loss function of the second phase, including: the DUN, the feature encoder LE$'$, and the diffusion denoising network $\boldsymbol{\epsilon}_\theta$. The objective loss function of the second phase can be stated as:

$$\mathcal{L}_{\text{diff}} = \|\hat{\boldsymbol{z}} - \boldsymbol{z}\|_1, \quad \mathcal{L}_{\text{all}} = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{diff}}. \tag{12}$$

Here, we do not use Eq. (11) as $\mathcal{L}_{\text{diff}}$ because it only trains diffusion at 't'-th step while we execute the all-time-step together and let the LDM directly predict $\hat{\boldsymbol{z}}$. The entire two-phase training procedure is summarized in Algorithm 1.

4.3 Aggregate Features by Trident Transformer

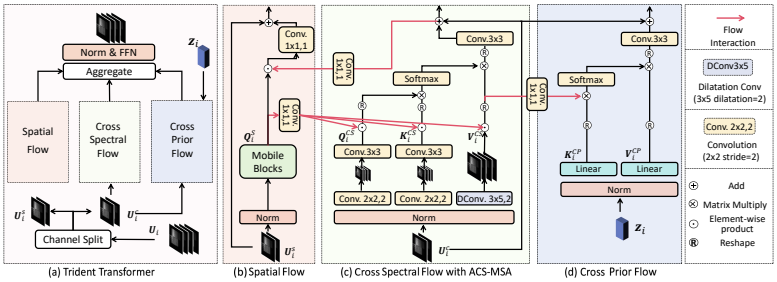

Fig. 4: (a) The Trident Transformer in Fig. 3(d). (b)-(d) are the detailed sub-modules. $\mathbf{U}_i$ is the input feature. The prior feature $\mathbf{Z}_i$ is sent into the prior flow.

Previous HSI reconstruction methods usually only exploit the relation between spatial and spectral, both externally and internally. However, the spatial-spectral relations are challenging to explore only with compressed measurements. Therefore, we design a Transformer, named Trident Transformer (TT), to effectively aggregate high-quality degradation-free prior knowledge for compensation.

Firstly, inspired by the multi-scale operations in previous papers [12,68] with hierarchical structures, we downsample the prior to obtain the multi-scale prior representations along with the U-shape levels in Fig. 3(d). Specifically, three

downsampling layers are employed, and the outputs contain prior features of three scales, stated as:

$$\boldsymbol{z}^i = \begin{cases} \boldsymbol{z}_{GT} \text{ or } \hat{\boldsymbol{z}}, & if\ i = 1, \\ \text{downsample}(\boldsymbol{z}^{i-1}), & if\ i > 1 \end{cases}, \tag{13}$$

where $\boldsymbol{z}^i \in \mathbb{R}^{\frac{N}{2^{i-1}} \times 2^{i-1}C}$, $i = 1, 2, 3$. For Phase I, $\boldsymbol{z}^1 = \boldsymbol{z}_{GT}$, which is computed in the first phase training; For Phase II, $\boldsymbol{z}^1 = \hat{\boldsymbol{z}}$, which is utilized for training and inference in the second phase.

As shown in Fig. 4, our Trident Transformer includes three branches: spatial flow, cross-spectral flow, and cross-prior flow. Each branch shares the information flow with others and is then fused by the aggregation layer and a feed-forward network (FFN). Before the embedding layer, the input feature at i-th scale $\mathbf{U}_i \in \mathbb{R}^{H_i \times W_i \times C_i}$ is split into $\mathbf{U}_i^C \in \mathbb{R}^{H_i \times W_i \times \frac{C_i}{2}}$ and $\mathbf{U}_i^S \in \mathbb{R}^{H_i \times W_i \times \frac{C_i}{2}}$ along the channel dimension, denoting cross flow input and spatial flow input respectively. The spatial flow consists of a series of MBlocks without batch norms.

Cross Spectral Flow In the cross spectral flow (CSF) module, as shown in Fig. 4 (c), we design asymmetric cross-scale multi-head self-attention (ACS-MHSA). Pansharpening (PAN) is a technique of using a high-resolution (HR) panchromatic and a lower-resolution (LR) HSI to generate an HR-HSI. Compared with directly capturing HR-HSI, it requires less amount of data. Inspired by the PAN, this flow primarily focuses on the spectral dimension and aims to save computational burden according to the spatial size. Specifically, we compress the spatial resolution of the query embedding ($\mathbf{Q}$) and key embedding ($\mathbf{K}$) to $\frac{1}{4}$ and expand its channel twice. Considering that CASSI measurements are shifted along one axis, there are more spatial correlations along this axis. Thus, after establishing the spectral correlation, we use an asymmetric dilation convolution (DConv) with kernel size 3×5 on the value embedding ($\mathbf{V}$) to obtain larger perceptual field information along the shifted axis with expanded channel dimension and unchanged spatial dimension. Embedding $\mathbf{Q}$, $\mathbf{K}$, $\mathbf{V}$, and spatial compensation $\mathbf{P}_i^S$ from spatial flow are formulated as:

$$\mathbf{Q}_i^{CS} = \mathbf{W}^{QCS}\mathbf{U}_i, \qquad \mathbf{K}_i^{CS} = \mathbf{W}^{KCS}\mathbf{U}_i, \qquad \mathbf{V}_i^{CS} = \mathbf{W}^{VCS}\mathbf{U}_i, \tag{14}$$

$$\mathbf{P}_i^{SQK} = \downarrow (\mathbf{W}^{PSQ}\mathbf{Q}_i^S), \qquad \mathbf{P}_i^{SV} = \mathbf{W}^{PSV}\mathbf{Q}_i^S, \qquad \mathbf{M}_i = \mathbf{Q}_i^{CS}(\mathbf{K}_i^{CS})^\top, \tag{15}$$

where $\mathbf{W}^*$ represents the weights of bias-free convolution, and $\downarrow$ is downsampling. The procedure of asymmetric cross-scale self-attention can be formulated as:

$$\text{ACS-MHSA}_i(\mathbf{U}_i) = \mathbf{P}_i^{SV} \odot \mathbf{W}_{c1}^{CS}\mathbf{V}_i^{CS}\text{Softmax}(\mathbf{P}_i^{SQK} \odot \mathbf{M}_i/\alpha), \tag{16}$$

Cross Prior Flow Cross prior flow (CPF) in Fig. 4 (d) is a variable shared multi-head cross-attention. The query in this flow is borrowed from the value of CSF which is extracted from a large perceptive field with more spatial information. In this way, the prior could facilitate to compensate for spatial deficiency. Compared to the spectral recovery, the spatial recovery is typically more chal-

lenging. Our manipulation can be formulated as:

$$MHSA_i^{CP}(\mathbf{U}_i) = W_{c1}\mathbf{V} \odot \text{Softmax}(\mathbf{K} \odot \mathbf{Q}/\alpha), \tag{17}$$

$$\mathbf{Q}_i^{CP} = \mathbf{Q}_i^{CS}, \quad \mathbf{K}_i^{CP} = \mathbf{W}_z^K \boldsymbol{z}_i, \quad \mathbf{V}_i^{CP} = \mathbf{W}_z^V \boldsymbol{z}_i, \tag{18}$$

where $\boldsymbol{z}^i, i = 1, 2, 3$ denotes the prior feature of different spatial levels.

Flow Interaction and Aggregation In order to compensate for the deficiency of spatial information in CSF and CPF, and the spectral information in the spatial flow, we fuse the compensation information together to reconstruct hyperspectra images. As shown in Fig. 4, the colorful arrows represent the information interactions between each flow. Specifically, information of each module modulates with other flows, where the 1×1 convolutions serve as compensation bridges. The aggregation part consists of concatenation, convolution layers, and an activation function (details can be seen in SM) for a weighted combination of each flow output. In our Trident Transformer, the prior knowledge learned from the clean images will provide compensation for reconstruction in both spatial and spectral details, avoiding the influence of degraded measurements.

Algorithm 1 Two-phase Training Strategy

Require: Dataset $\mathcal{D} = \{(\boldsymbol{x}^{(n)}, \boldsymbol{y}^{(n)}\}_{n=1}^N$; Sensing matrix $\boldsymbol{A}$; Random initialized parameter of DUN ϕ_{DUN}, latent encoder LE network ϕ_{LE}, conditional encoder LE' network ϕ_{CLE}, diffusion denoising network ϕ_ϵ;

1: **while** Not Converge **do** ▷ Phase I training, '←' denotes update
2: $\boldsymbol{z}_{GT} \leftarrow \text{LE}(\boldsymbol{I}_{\text{E}}|\phi_{LE})$; $\hat{x} \leftarrow \text{DUN}(\boldsymbol{y}, \boldsymbol{A}, \boldsymbol{z}_{GT}|\phi_{DUN})$
3: Jointly update ϕ_{DUN} and ϕ_{LE} by $\mathcal{L}_{rec}$;
4: Freeze ϕ_{LE};
5: **while** Not Converge **do** ▷ Phase II training
6: $\boldsymbol{c} \leftarrow \text{LE}'(\boldsymbol{y}_{norm}|\phi_{CLE})$; $\hat{\boldsymbol{z}} \leftarrow \text{Diff}(\boldsymbol{c}|\phi_\epsilon)$;
7: $\boldsymbol{z}_{GT} \leftarrow \text{LE}(\boldsymbol{I}_{\text{E}}|\phi_{LE})$; $\hat{x} \leftarrow \text{DUN}(\boldsymbol{y}, \boldsymbol{A}, \hat{\boldsymbol{z}})|\phi_{DUN})$;
8: Jointly update $\phi_{DUN}, \phi_{CLE},$ and ϕ_ϵ by $\mathcal{L}_{all}$ in Eq. (12);

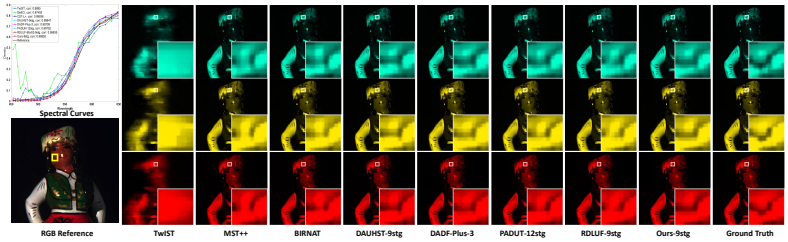

Fig. 5: The visualization result on synthetic data. 3 out of 28 wavelengths are selected for visual comparison. 'Corr' in the top left curve is the correlation coefficient between one method curve and the ground truth curve of the chosen (golden box) region.

Table 1: The results of PSNR in dB (top entry in each cell), SSIM (bottom entry in each cell) on the 10 synthetic spectral scenes.'-3stg' denotes the network with 3 unfolding stages. 'Avg' represents the average of 10 scenes. **Bold**: Best.

Algorithms	Scene1	Scene2	Scene3	Scene4	Scene5	Scene6	Scene7	Scene8	Scene9	Scene10	Avg
TwIST	25.16	23.02	21.40	30.19	21.41	20.95	22.20	21.82	22.42	22.67	23.12
	0.700	0.604	0.711	0.851	0.635	0.644	0.643	0.650	0.690	0.569	0.669
DNU	31.72	31.13	29.99	35.34	29.03	30.87	28.99	30.13	31.03	29.14	30.74
	0.863	0.846	0.845	0.908	0.833	0.887	0.839	0.885	0.876	0.849	0.863
MST++	35.40	35.87	36.51	42.27	32.77	34.80	33.66	32.67	35.39	32.50	35.99
	0.941	0.944	0.953	0.973	0.947	0.955	0.925	0.948	0.949	0.941	0.951
BIRNAT	36.79	37.89	40.61	46.94	35.42	35.30	36.58	33.96	39.47	32.80	37.58
	0.951	0.957	0.971	0.985	0.964	0.959	0.955	0.956	0.970	0.938	0.960
LRSDN	35.44	34.89	38.90	45.29	34.71	33.18	37.76	30.57	39.49	30.62	36.08
	0.923	0.909	0.961	0.985	0.949	0.930	0.964	0.901	0.963	0.889	0.938
DAUHST-9stg	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
DADF-Plus-3	37.46	39.86	41.03	45.98	35.53	37.02	36.76	34.78	40.07	34.39	38.29
	0.965	0.976	0.974	0.989	0.972	0.975	0.958	0.971	0.976	0.962	0.972
PADUT-5stg	36.68	38.74	41.37	45.79	35.13	36.37	36.52	34.40	39.57	33.78	37.84
	0.955	0.969	0.975	0.988	0.967	0.969	0.959	0.967	0.971	0.955	0.967
RDLUF-MixS2-3stg	36.67	38.48	40.63	46.04	34.63	36.18	35.85	34.37	38.98	33.73	37.56
	0.953	0.965	0.971	0.986	0.963	0.966	0.951	0.963	0.966	0.950	0.963
Ours-3stg	37.14	39.60	41.78	46.57	35.57	37.02	36.80	35.22	40.15	34.17	38.31
	0.963	0.975	0.978	0.990	0.971	0.975	0.960	0.973	0.976	0.962	0.972
Ours-5stg	37.88	40.92	43.41	47.18	37.12	37.74	38.28	35.73	41.48	35.18	39.38
	0.968	0.980	0.983	0.992	0.978	0.980	0.969	0.977	0.981	0.967	0.977
PADUT-12stg	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-MixS2-9stg	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
Ours-9stg	**38.08**	41.84	43.77	47.99	37.97	38.30	38.82	36.15	42.53	35.48	40.09
	0.969	0.982	0.983	**0.993**	0.980	0.980	0.973	**0.979**	**0.984**	**0.970**	0.979
Ours-10stg	**38.08**	**41.85**	**43.83**	**48.04**	**38.00**	**38.32**	**38.94**	**36.20**	**42.81**	**35.54**	**40.16**
	0.970	**0.984**	**0.984**	**0.993**	**0.982**	**0.982**	**0.974**	**0.979**	**0.984**	**0.970**	**0.980**

Table 2: Performance and computational efficiency comparisons with recent methods.

Method	Venue	PSNR (dB)	Params (M)	FLOPs (G)	Infer. Time (ms)	Training Time (h)
PADUT-12stg	ICCV'23	38.89	5.38	90.46	**749.94**	**123.3**
RDLUF-9stg	CVPR'23	39.57	**1.89**	231.09	913.34	155.5
Ours-9stg	-	**40.09**	2.78	**88.68**	1096.58	143.6

Table 3: Ablation study of different modules in our 3-stage unfolding network. 'PSNR' is the average of 10 synthetic scenes. 'FLOPs (G)' denotes the FLOPs in testing.

Method	Projection	SF	CSF Atten.	CPF	LE Input	Diffusion	PSNR	FLOPs (G)
w/o GC-GAP	Basic GAP	✓	ACS	✓	$\boldsymbol{I}_{\mathrm{E}}$	✓	38.01	29.84
w/o spatial flow	GC-GAP	✗	ACS	✓	$\boldsymbol{I}_{\mathrm{E}}$	✓	37.66	30.13
Basic MHSA	GC-GAP	✓	Basic	✓	$\boldsymbol{I}_{\mathrm{E}}$	✓	38.31	36.80
w/o prior flow	GC-GAP	✓	ACS	✗	$\boldsymbol{I}_{\mathrm{E}}$	✓	37.89	30.37
Inaccurate guidance	GC-GAP	✓	ACS	✓	$\boldsymbol{y}$	✓	37.63	33.80
w/o diffusion	GC-GAP	✓	ACS	✓	$\boldsymbol{I}_{\mathrm{E}}$	✗	37.61	33.70
Our full model	GC-GAP	✓	ACS	✓	$\boldsymbol{I}_{\mathrm{E}}$	✓	38.40	33.80

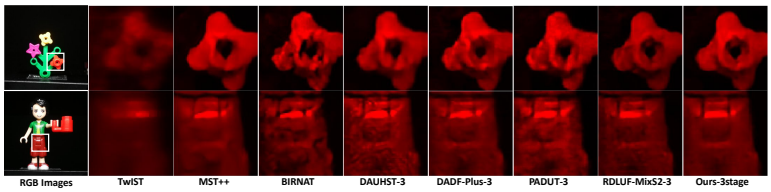

Fig. 6: The real data visual comparisons. 2 out of 28 wavelengths are selected.

5 Experiments

We conduct experiments on both simulation and real HSI datasets. Following the approaches in [5, 26, 43, 45], we select a set of 28 wavelengths ranging from 450-650nm by employing spectral interpolation applied to the HSI data.

5.1 Experimental Settings

Simulation and Real Datasets: We adopt two widely used HSI datasets, i.e., CAVE [48] and KAIST [14] for simulation experiments. The CAVE dataset comprises 32 HSIs with a spatial size of 512 × 512. The KAIST dataset includes 30 HSIs with a spatial size of 2704 × 3376. Following previous works [5,26,43,45], we only employ the CAVE dataset as the training set both in Phase I and II, while 10 scenes from the KAIST dataset are utilized for testing. During the training process, a real mask of size 256 × 256 pixels is applied. In our real experiment, we utilized the HSI dataset captured by the SD-CASSI system in [45]. The system captures real-world scenes of size 660 × 714× 28 with wavelengths spanning from 450 to 650 nm and dispersion of 54 pixels.

Implementation Details: For the diffusion settings, the iteration number T of the diffusion is set to 16, and the latent space dimension N is set to 16. Training Phase I (train DUN and LE) needs 200 epochs and Phase II (train DU, $\boldsymbol{\epsilon}_\theta$, and LE') needs 100 epochs. For all phases of training, we use the Adam [27] optimizer and cosine scheduler. PSNR and SSIM [67] are utilized as our metrics. Our method is implemented with the PyTorch and trained using NVIDIA RTX3090 GPUs. More details can be seen in the supplementary material (SM).

5.2 Compare with State-of-the-art

We compare our method with previous methods including the end-to-end networks: DADF-Net [69], MST [5], BIRNAT [13]; the deep unfolding methods: RDLUF-MixS2 [15], PADUT [32], DAUHST [6], DNU [65]; the self-supervised method: LRSDN [11]; and traditional model-based method: TwIST [2]. The comparisons are conducted on both synthetic and real datasets.

Synthetic data: The numeric comparisons on synthetic data can be seen in Table 1. Our proposed method surpasses the recent SOTA method RDLUF-MixS2 according to average PSNR (+0.59 dB) and SSIM (+0.06). Fig. 5 shows

the visual reconstruction results. Three wavelengths including striking colors in RGB reference red, yellow, and green are selected to compare. The golden box part in the reference was chosen to calculate and compare the wavelength accuracy. The accuracy metric is the correlation coefficient with the ground truth of the chosen region, *i.e.* the 'Corr' in the curves. According to the 'Corr', our method (0.9995) has a more accurate wavelength curve than others. The zoomed part in the figure also demonstrates that our method has clearer edges on the hat than others. In Table. 2, we list the performance, parameter number, FLOPs, inference time, and training time of our method and other recent unfolding methods. 'Infer. Time' denotes the total inference time of each method dealing with 10 synthetic test scenes. 'Training Time' is under 300 total epochs setting. The table illustrates that our methods exhibit superior performance, the lowest FLOPs, along with reasonable parameter counts and times. **Real data:** Two scenes of real SD-CASSI measurement reconstruction results are shown in Fig. 6, and two obvious color regions in RGB references are selected to compare. Our method shows less noise and artefacts on the plastic toy surface.

6 Ablation Study

In this ablation study, we train our model on the synthetic training data with models with 3 unfolding stages. The results are summarized in Table 3. 'Inaccurate guidance' denotes that we only use measurement as the latent encoder input instead of clean HSIs. 'w/o prior flow' denotes that using simple MLP layers instead of CPF. 'w/o diffusion' denotes removing diffusion in Phase II.

The ablation illustrates that with LDM prior assistance, we can achieve better reconstruction results, and our design of the Trident Transformer successfully aggregates three types of information and effectively compensates for some reconstruction defects. Fig. 2 also visualizes the feature map changes before and after prior enhancement. The enhanced features demonstrate increased concentration on edges and reduced noise. Moreover, without accurate guidance, the LDM will even harm the reconstruction. The method 'Basic MHSA' illustrates that ACS-MHSA in CSF has better performance and higher computational efficiency than basic MHSA. The diffusion steps shown in Fig. 1(b) illustrates that 16 steps are enough for good reconstruction results. Table 2 illustrates that inference time is still in a reasonable range even using diffusion 16 steps. For the scale of $\boldsymbol{z}$, we compared different values of N in Table 4, and we find that $N = 16$ can keep the balance of performance and efficiency. More ablation studies can be seen in SM.

$\boldsymbol{z}$ scale (N)	4	16	64	256
PSNR	37.99	38.40	38.40	38.41
FLOPs	33.31	33.80	34.49	36.99

Table 4: $\boldsymbol{z}$ scale comparisons.

7 Conclusion

In this paper, we introduce a novel deep unfolding network that leverages prior knowledge from the latent diffusion model for spectral reconstruction. It achieves SOTA performance on both simulated data and real data.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (grant number 62271414), Zhejiang Provincial Outstanding Youth Science Foundation (grant number LR23F010001), Zhejiang "Pioneer" and "Leading Goose" R&D Program (grant number 2024SDXHDX0006, 2024C03182), the Key Project of Westlake Institute for Optoelectronics (grant number 2023GD007), and the 2023 International Sci-tech Cooperation Projects under the purview of the "Innovation Yongjiang 2035" Key R&D Program (grant number 2024Z126).

References

1. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Img. Sci. **2**(1), 183–202 (2009)
2. Bioucas-Dias, J., Figueiredo, M.: A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration. IEEE TIP **16**(12), 2992–3004 (2007)
3. Blattmann, A., Rombach, R., Ling, H., Dockhorn, T., Kim, S.W., Fidler, S., Kreis, K.: Align your latents: High-resolution video synthesis with latent diffusion models. In: CVPR. pp. 22563–22575 (2023)
4. Cai, Y., Lin, J., Hu, X., Wang, H., Yuan, X., Zhang, Y., Timofte, R., Van Gool, L.: Coarse-to-fine sparse transformer for hyperspectral image reconstruction. In: ECCV. pp. 686–704 (2022)
5. Cai, Y., Lin, J., Hu, X., Wang, H., Yuan, X., Zhang, Y., Timofte, R., Van Gool, L.: Mask-guided spectral-wise transformer for efficient hyperspectral image reconstruction. In: CVPR. pp. 17502–17511 (2022)
6. 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. NeurIPS pp. 37749–37761 (2022)
7. Cai, Y., Zheng, Y., Lin, J., Yuan, X., Zhang, Y., Wang, H.: Binarized spectral compressive imaging. NeurIPS **36** (2024)
8. Chan, S.H., Wang, X., Elgendy, O.A.: Plug-and-play ADMM for image restoration: Fixed-point convergence and applications. IEEE TCI **3**, 84–98 (2017)
9. Charles, A.S., Olshausen, B.A., Rozell, C.J.: Learning sparse codes for hyperspectral imagery. IEEE JSTSP **5**(5), 963–978 (2011)
10. Chen, Y., Gui, X., Zeng, J., Zhao, X.L., He, W.: Combining low-rank and deep plug-and-play priors for snapshot compressive imaging. IEEE TNNLS (2023)
11. Chen, Y., Lai, W., He, W., Zhao, X.L., Zeng, J.: Hyperspectral compressive snapshot reconstruction via coupled low-rank subspace representation and self-supervised deep network. IEEE TIP (2024)
12. Chen, Z., Zhang, Y., Liu, D., Xia, B., Gu, J., Kong, L., Yuan, X.: Hierarchical integration diffusion model for realistic image deblurring. arXiv preprint arXiv:2305.12966 (2023)
13. Cheng, Z., Chen, B., Lu, R., Wang, Z., Zhang, H., Meng, Z., Yuan, X.: Recurrent neural networks for snapshot compressive imaging. IEEE TPAMI **45**(2), 2264–2281 (2022)
14. Choi, I., Jeon, D.S., Nam, G., Gutierrez, D., Kim, M.H.: High-quality hyperspectral reconstruction using a spectral prior. ACM TOG **36**(6), 218:1–13 (2017)

15. 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)
16. Gao, S., Liu, X., Zeng, B., Xu, S., Li, Y., Luo, X., Liu, J., Zhen, X., Zhang, B.: Implicit diffusion models for continuous super-resolution. In: CVPR. pp. 10021–10030 (2023)
17. Gehm, M.E., John, R., Brady, D.J., Willett, R.M., Schulz, T.J.: Single-shot compressive spectral imaging with a dual-disperser architecture. Optics Express **15**(21), 14013–14027 (2007)
18. Goetz, A.F., Vane, G., Solomon, J.E., Rock, B.N.: Imaging spectrometry for earth remote sensing. science **228**(4704), 1147–1153 (1985)
19. Harvey, W., Naderiparizi, S., Masrani, V., Weilbach, C., Wood, F.: Flexible diffusion modeling of long videos. NeurIPS **35**, 27953–27965 (2022)
20. Hendrycks, D., Gimpel, K.: Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415 (2016)
21. Ho, J., Chan, W., Saharia, C., Whang, J., Gao, R., Gritsenko, A., Kingma, D.P., Poole, B., Norouzi, M., Fleet, D.J., et al.: Imagen video: High definition video generation with diffusion models. arXiv preprint arXiv:2210.02303 (2022)
22. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. NeurIPS **33**, 6840–6851 (2020)
23. Höppe, T., Mehrjou, A., Bauer, S., Nielsen, D., Dittadi, A.: Diffusion models for video prediction and infilling. arXiv preprint arXiv:2206.07696 (2022)
24. Howard, A., Sandler, M., Chu, G., Chen, L.C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., et al.: Searching for mobilenetv3. In: ICCV. pp. 1314–1324 (2019)
25. Hu, X., Cai, Y., Lin, J., Wang, H., Yuan, X., Zhang, Y., Timofte, R., Van Gool, L.: Hdnet: High-resolution dual-domain learning for spectral compressive imaging. In: CVPR. pp. 17542–17551 (2022)
26. 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)
27. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
28. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. CoRR **abs/1312.6114** (2013)
29. 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)
30. Lai, Z., Fu, Y., Zhang, J.: Hyperspectral image super resolution with real unaligned rgb guidance. IEEE TNNLS (2024)
31. Li, L., Li, W., Qu, Y., Zhao, C., Tao, R., Du, Q.: Prior-based tensor approximation for anomaly detection in hyperspectral imagery. IEEE TNNLS **33**(3), 1037–1050 (2020)
32. Li, M., Fu, Y., Liu, J., Zhang, Y.: Pixel adaptive deep unfolding transformer for hyperspectral image reconstruction. In: ICCV. pp. 12959–12968 (2023)
33. Li, M., Fu, Y., Zhang, Y.: Spatial-spectral transformer for hyperspectral image denoising. In: AAAI. vol. 37, pp. 1368–1376 (2023)
34. Li, M., Liu, J., Fu, Y., Zhang, Y., Dou, D.: Spectral enhanced rectangle transformer for hyperspectral image denoising. In: CVPR. pp. 5805–5814 (2023)
35. Li, S., Song, W., Fang, L., Chen, Y., Ghamisi, P., Benediktsson, J.A.: Deep learning for hyperspectral image classification: An overview. IEEE TGRS **57**(9), 6690–6709 (2019)

36. Liao, X., Li, H., Carin, L.: Generalized alternating projection for weighted-2,1 minimization with applications to model-based compressive sensing. SIAM Journal on Imaging Sciences **7**(2), 797–823 (2014)
37. Liu, Y., Yuan, X., Suo, J., Brady, D., Dai, Q.: Rank minimization for snapshot compressive imaging. IEEE TPAMI **41**(12), 2990–3006 (Dec 2019)
38. Lu, G., Fei, B.: Medical hyperspectral imaging: a review. Journal of biomedical optics **19**(1), 010901–010901 (2014)
39. Lu, R., Chen, B., Cheng, Z., Wang, P.: Rafnet: Recurrent attention fusion network of hyperspectral and multispectral images. Signal Processing **177**, 107737 (2020)
40. Lu, R., Chen, B., Sun, J., Chen, W., Wang, P., Chen, Y., Liu, H., Varshney, P.K.: Heterogeneity-aware recurrent neural network for hyperspectral and multispectral image fusion. IEEE JSTSP **16**(4), 649–665 (2022)
41. Lugmayr, A., Danelljan, M., Romero, A., Yu, F., Timofte, R., Van Gool, L.: Repaint: Inpainting using denoising diffusion probabilistic models. In: CVPR. pp. 11461–11471 (2022)
42. Ma, J., Liu, X.Y., Shou, Z., Yuan, X.: Deep tensor admm-net for snapshot compressive imaging. In: ICCV. pp. 10223–10232 (2019)
43. Meng, Z., Jalali, S., Yuan, X.: Gap-net for snapshot compressive imaging. arXiv preprint arXiv:2012.08364 (2020)
44. Meng, Z., Ma, J., Yuan, X.: End-to-end low cost compressive spectral imaging with spatial-spectral self-attention. In: ECCV. pp. 187–204 (2020)
45. Meng, Z., Ma, J., Yuan, X.: End-to-end low cost compressive spectral imaging with spatial-spectral self-attention. In: ECCV (2020)
46. Meng, Z., Yuan, X., Jalali, S.: Deep unfolding for snapshot compressive imaging. International Journal of Computer Vision pp. 1–26 (2023)
47. Miao, X., Yuan, X., Pu, Y., Athitsos, V.: λ-net: Reconstruct hyperspectral images from a snapshot measurement. In: ICCV (2019)
48. Park, J.I., Lee, M.H., Grossberg, M.D., Nayar, S.K.: Multispectral imaging using multiplexed illumination. In: 2007 IEEE 11th ICCV. pp. 1–8. IEEE (2007)
49. Qiu, H., Wang, Y., Meng, D.: Effective snapshot compressive-spectral imaging via deep denoising and total variation priors. In: CVPR. pp. 9127–9136 (2021)
50. Rao, W., Gao, L., Qu, Y., Sun, X., Zhang, B., Chanussot, J.: Siamese transformer network for hyperspectral image target detection. IEEE TGRS **60**, 1–19 (2022)
51. ul Rehman, A., Qureshi, S.A.: A review of the medical hyperspectral imaging systems and unmixing algorithms' in biological tissues. Photodiagnosis and Photodynamic Therapy **33**, 102165 (2021)
52. Ren, M., Delbracio, M., Talebi, H., Gerig, G., Milanfar, P.: Multiscale structure guided diffusion for image deblurring. In: ICCV. pp. 10721–10733 (2023)
53. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR. pp. 10684–10695 (2022)
54. Ryu, E., Liu, J., Wang, S., Chen, X., Wang, Z., Yin, W.: Plug-and-play methods provably converge with properly trained denoisers. In: ICML. pp. 5546–5557 (2019)
55. Saharia, C., Ho, J., Chan, W., Salimans, T., Fleet, D.J., Norouzi, M.: Image super-resolution via iterative refinement. IEEE TPAMI **45**(4), 4713–4726 (2022)
56. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: ICML. pp. 2256–2265 (2015)
57. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502 (2020)

58. Tolstikhin, I.O., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., et al.: Mlp-mixer: An all-mlp architecture for vision. NeurIPS pp. 24261–24272 (2021)
59. Uzkent, B., Rangnekar, A., Hoffman, M.: Aerial vehicle tracking by adaptive fusion of hyperspectral likelihood maps. In: CVPR Workshops. pp. 39–48 (2017)
60. Van Nguyen, H., Banerjee, A., Chellappa, R.: Tracking via object reflectance using a hyperspectral video camera. In: CVPR Workshops. pp. 44–51 (2010)
61. Wagadarikar, A., John, R., Willett, R., Brady, D.: Single disperser design for coded aperture snapshot spectral imaging. Applied optics **47**(10), B44–B51 (2008)
62. Wagadarikar, A., John, R., Willett, R., Brady, D.: Single disperser design for coded aperture snapshot spectral imaging. Applied Optics **47**(10), B44–B51 (2008)
63. 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 (2017)
64. Wang, L., Wu, Z., Zhong, Y., Yuan, X.: Snapshot spectral compressive imaging reconstruction using convolution and contextual transformer. Photonics Research **10**(8), 1848–1858 (2022)
65. Wang, L., Sun, C., Zhang, M., Fu, Y., Huang, H.: Dnu: Deep non-local unrolling for computational spectral imaging. In: CVPR. pp. 1661–1671 (2020)
66. Wang, Z., Chen, B., Lu, R., Zhang, H., Liu, H., Varshney, P.K.: Fusionnet: An unsupervised convolutional variational network for hyperspectral and multispectral image fusion. IEEE TIP **29**, 7565–7577 (2020)
67. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE TIP **13**(4), 600–612 (2004)
68. Xia, B., Zhang, Y., Wang, S., Wang, Y., Wu, X., Tian, Y., Yang, W., Van Gool, L.: Diffir: Efficient diffusion model for image restoration. arXiv preprint arXiv:2303.09472 (2023)
69. Xu, P., Liu, L., Zheng, H., Yuan, X., Xu, C., Xue, L.: Degradation-aware dynamic fourier-based network for spectral compressive imaging. IEEE TMM (2023)
70. Yuan, X.: Generalized alternating projection based total variation minimization for compressive sensing. In: ICIP. pp. 2539–2543 (2016)
71. Yuan, X., Liu, Y., Suo, J., Dai, Q.: Plug-and-play algorithms for large-scale snapshot compressive imaging. In: CVPR (June 2020)
72. Yuan, X., Liu, Y., Suo, J., Durand, F., Dai, Q.: Plug-and-play algorithms for video snapshot compressive imaging. IEEE TPAMI pp. 1–1 (2021)
73. Zhang, Q., Chen, Y.: Fast sampling of diffusion models with exponential integrator. arXiv preprint arXiv:2204.13902 (2022)
74. 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)
75. Zhang, T., Fu, Y., Li, C.: Hyperspectral image denoising with realistic data. In: ICCV. pp. 2248–2257 (2021)
76. Zhang, T., Liang, Z., Fu, Y.: Joint spatial-spectral pattern optimization and hyperspectral image reconstruction. IEEE JSTSP **16**(4), 636–648 (2022)