sensors

Article

Hybrid Sparse Transformer and Wavelet Fusion-Based Deep Unfolding Network for Hyperspectral Snapshot Compressive Imaging

Yangke Ying [1], **Jin Wang** [2], **Yunhui Shi** [1,*] and **Nam Ling** [3]

[1] Beijing Key Laboratory of Multimedia and Intelligent Software Technology, Beijing Institute of Artificial Intelligence, School of Information Science and Technology, Beijing University of Technology, Beijing 100124, China; yingyangke@emails.bjut.edu.cn
[2] Beijing Institute of Artificial Intelligence, School of Computer Science, Beijing University of Technology, Beijing 100124, China; ijinwang@bjut.edu.cn
[3] Department of Computer Science and Engineering, Santa Clara University, Santa Clara, CA 95053, USA; nling@scu.edu
* Correspondence: syhzm@bjut.edu.cn

Abstract: Recently, deep unfolding network methods have significantly progressed in hyperspectral snapshot compressive imaging. Many approaches directly employ Transformer models to boost the feature representation capabilities of algorithms. However, they often fall short of leveraging the full potential of self-attention mechanisms. Additionally, current methods lack adequate consideration of both intra-stage and inter-stage feature fusion, which hampers their overall performance. To tackle these challenges, we introduce a novel approach that hybridizes the sparse Transformer and wavelet fusion-based deep unfolding network for hyperspectral image (HSI) reconstruction. Our method includes the development of a spatial sparse Transformer and a spectral sparse Transformer, designed to capture spatial and spectral attention of HSI data, respectively, thus enhancing the Transformer's feature representation capabilities. Furthermore, we incorporate wavelet-based methods for both intra-stage and inter-stage feature fusion, which significantly boosts the algorithm's reconstruction performance. Extensive experiments across various datasets confirm the superiority of our proposed approach.

Keywords: compressive sensing; hyperspectral image reconstruction; snapshot compressive imaging; deep unfolding network

check for updates

Citation: Ying, Y.; Wang, J.; Shi, Y.; Ling, N. Hybrid Sparse Transformer and Wavelet Fusion-Based Deep Unfolding Network for Hyperspectral Snapshot Compressive Imaging. *Sensors* **2024**, *24*, 6184. https://doi.org/10.3390/s24196184

Academic Editor: Subrata Chakraborty

Received: 13 August 2024
Revised: 20 September 2024
Accepted: 21 September 2024
Published: 24 September 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The continuous demand for capturing high-dimensional data has driven the development of imaging devices and processing algorithms. Compared to imaging systems that acquire RGB images, hyperspectral imaging (HSI) systems can capture a wider range of wavelength information from scenes. As a result, they have rapidly developed and been applied to various downstream visual tasks [1–6]. Traditional HSI systems primarily acquire scene information through one-dimensional or two-dimensional scanning mechanisms [7]. However, these systems have slow imaging speeds due to the need for multiple exposures, making them unsuitable for dynamic scenes. In recent years, many snapshot compressive imaging (SCI) systems [8–10] have emerged with the advancement of compressive sensing theory [11]. These systems can obtain three-dimensional HSI data from two-dimensional observations with a single exposure. Among SCI systems, coded aperture snapshot spectral imaging (CASSI) systems [9] have gained widespread use due to their advantages of low cost, low power consumption, and high sampling rates. Specifically, CASSI systems consist of two main components: a hardware encoder and a software decoder. The hardware encoder uses optical devices like coded apertures and prisms to modulate three-dimensional

HSI scenes into two-dimensional compressed measurements. The primary task of the software decoder is to reconstruct the original three-dimensional HSI scenes from the acquired two-dimensional measurements. The quality of data reconstruction entirely depends on the effectiveness of the algorithm used. Therefore, the core challenge of this research is to reconstruct higher-quality original HSI data.

Current reconstruction algorithms can be broadly categorized into two types: traditional model-based optimization methods and learning-based deep neural network methods. Model-based methods typically use various handcrafted priors to solve traditional ill-posed optimization inverse problems [12–20]. While these methods benefit from mathematical derivations, enhancing their theoretical interpretability, they face limitations in prior design and often exhibit slow reconstruction speeds. On the other hand, deep learning techniques, with their powerful feature modeling capabilities, have shown outstanding performance in many tasks. Consequently, many learning-based algorithms have been applied to HSI reconstruction to overcome the limitations of model-based methods. Learning-based methods can be further divided into three strategies: end-to-end (E2E) methods, plug-and-play (PnP) methods, and deep unfolding network (DUN) methods. E2E methods [21–28]. use deep networks to directly establish a mapping between three-dimensional HSI data and two-dimensional compressed measurements. Although E2E methods have achieved some impressive reconstruction results, they still suffer from a lack of interpretability due to the black-box nature of the networks. PnP methods [29–33] integrate deep network modules into model-based methods, replacing traditional handcrafted priors with deep network modules. While PnP methods more effectively address the prior subproblem in optimization models, they still follow the traditional optimization process and do not fundamentally overcome the limitations of model-based methods. DUN methods [34–50] construct multi-stage unfolding networks to implement the iterative solving process of model-based methods in an end-to-end manner. DUN methods alleviate the interpretability challenges and provide encouraging experimental results. As a result, DUN methods are rapidly evolving and showing great potential.

Despite the promising reconstruction results, existing DUN methods still face several challenges in HSI reconstruction. On the one hand, many existing DUN methods utilize the non-local information modeling capabilities of Transformer [51] modules to significantly enhance the final reconstruction performance. Existing Transformers typically model the correlations among all tokens in the query–key pairs. However, in practice, if some tokens in the query do not correlate with tokens in the key, the estimated self-attention values for these tokens are still used for feature aggregation, thereby limiting the final feature representation capability. Additionally, sparse self-attention mechanisms have demonstrated outstanding performance in numerous RGB image processing tasks [52,53]. On the other hand, DUN methods experience feature information loss within each network stage due to cross-scale transformations, as well as across different stages due to frequent signal-to-feature conversions. In existing methods, Herosnet [42] mitigates cross-stage information loss by concatenating features from earlier stages and passing them into the next stage. PADUT [45] achieves cross-stage information fusion by applying Fourier transform operations to the features from a frequency-domain perspective. RDLUF [46] reduces intra-stage multi-scale information loss through the convolutional fusion of features at different scales, while also minimizing cross-stage information loss by generating modulation parameters from previous stage features to guide information capture in the next stage. SUCTNet [54] introduces a dual transformer-based module to simultaneously utilize HSI interactions and details at both global and local scales. EDUNet! [49] enhances cross-stage feature transfer efficiency by constructing a cross-stage spectral self-attention network module that leverages HSI characteristics. M2U-Net [55] achieves feature fusion for the HSI super-resolution task through a cross-attention guided module. uHNTC [56] designs a multi-level cross-feature attention mechanism to achieve hierarchical spatial–spectral feature fusion for the hyperspectral and multispectral image fusion task. Recently, wavelet-based feature fusion methods have been applied to many low-level vision tasks [57,58] and have achieved impressive results. This prompts us to explore how to leverage the properties of the wavelet transform [59] to construct a feature fusion module for HSI data, aimed at addressing the issue of intra-stage and cross-stage feature information loss.

To address these challenges, we propose a novel method that hybridizes the sparse Transformer and wavelet fusion-based deep unfolding network for hyperspectral snapshot compressive imaging. Specifically, we introduce a sparse spatial Transformer and a sparse spectral Transformer to model the self-attention in the spatial and spectral dimensions, respectively. By extracting the most relevant regions through sparse operations to compute similarity for feature aggregation, we enhance the feature representation capability of the Transformer. Additionally, we utilize the cross-scale properties of the wavelet transform to construct a wavelet-based intra-stage feature fusion module, addressing the intra-stage feature information loss in DUN methods. Finally, we further leverage the wavelet transform to build an inter-stage feature fusion module, enabling cross-stage feature transmission in the wavelet domain and avoiding the information loss caused by frequent signal-to-feature conversions. The main contributions of this study can be summarized as follows:

- We propose a novel method that hybridizes the sparse Transformer and wavelet fusion-based deep unfolding network for hyperspectral snapshot compressive imaging.
- To enhance the expressive capability of existing Transformer methods, we propose a sparse spatial–spectral Transformer. This approach uses sparse operations to avoid calculating correlations for irrelevant tokens during feature aggregation.
- To address the issue of information loss within and across stages of the DUN method, we design the wavelet-based intra-stage fusion module and wavelet-based inter-stage fusion module, respectively. These fusion modules utilize the characteristics of the wavelet transform to enhance HSI reconstruction.

2. Related Works

2.1. Model-Based Traditional Optimization Methods

For reconstruction methods in the hyperspectral image snapshot compressive imaging task, traditional model-based optimization approaches incorporate various hand-crafted prior knowledge as regularization terms and solve the ill-posed inverse problem through iterative optimization. For instance, GAP-TV [12] incorporates the total variation (TV) prior term and utilizes the generalized alternating projection (GAP) [60] algorithm to address the optimization problem for HSI reconstruction. To exploit the sparse constraint of HSI for reconstruction, Lin et al. [13] suggests learning an overcomplete dictionary that represents HSI more sparsely than previous methods. The paper [14] leverages the spatial and spectral sparsity properties of HSI data to develop a dictionary learning approach based on sparse priors for HSI scene reconstruction. The paper [15] compares the impact of various estimation algorithms on the effectiveness of HSI reconstruction. CT3D [16] introduces a method for reconstructing HSI using coupled tensor decomposition and the alternating direction method of multipliers (ADMM) [61] iteration. MMLE-GMM [17] extends a maximum marginal likelihood estimator to a Gaussian mixture model with a primarily low-rank covariance matrix, achieving accelerated optimization for reconstruction. DeSCI [18] creates a joint model that integrates non-local self-similarity and rank minimization methods with the SCI sensing process, resulting in excellent reconstruction outcomes. NGmeet [20] proposes an algorithm that combines the global spectral low-rank property and spatial non-local self-similarity prior for hyperspectral image reconstruction. Traditional model-based iterative optimization methods, relying on limited prior designs and requiring multiple iterations for optimization, lead to slow reconstruction speeds and suboptimal outcomes.

2.2. Learning-Based Neural Network Methods

Due to the powerful non-linear feature modeling capabilities of deep neural networks, many learning-based methods are being employed to address the reconstruction of HSI data. These learning-based methods can be categorized into three strategies: (1) end-to-end (E2E) methods; (2) plug-and-play (PnP) methods; and (3) deep unfolding network (DUN) methods.

(1) The E2E methods directly model the mapping relationship between 2D compressed measurements and 3D HSI data through end-to-end learning using deep networks. For instance, λ-net [23] sets up a two-stage network, where the initial reconstruction stage utilizes self-attention for reconstruction, followed by a refinement stage to further improve the results. TSA-Net [24] tackles different dimensions separately through the use of spatial–spectral self-attention mechanisms. MST [25] introduces an innovative Mask-guided Spectral-wise Transformer designed for reconstructing HSI. HDNet [26] integrates dual-domain constraints in frequency and spatial–spectral domains within its objective function to improve the quality of reconstruction outcomes. CST [27] introduces a novel coarse-to-fine sparse Transformer approach that incorporates the sparsity of hyperspectral imaging into deep learning for reconstruction purposes. BIRNAT [28] integrates the expressive capabilities of an end-to-end convolutional framework with bidirectional recurrent neural networks (RNNs) to effectively capture sequence correlations in snapshot compressive imaging. However, the end-to-end (E2E) methods encounter challenges in interpretability because of the opaque nature of convolutional networks.

(2) The PnP methods often use the outcomes from pre-trained denoising networks to replace solving the prior subproblem in traditional model-based approaches. PnP-HSI [29] employs a deep learning denoising network as a regularization prior and addresses the reconstruction optimization problem using the ADMM algorithm. PnP-DIP-HSI [30] integrates a deep image prior (DIP) network as a prior within the iterative optimization algorithm, establishing a self-supervised framework for reconstructing HSI. Qiu et al. [31] extends the PnP algorithm by incorporating deep image denoising and total variation priors into the conventional optimization objective. LR2DP [32] and LRSDN [33] leverage the robust spectral correlation and complex spatial structures inherent in HSI for SCI reconstruction, integrating model-driven low-rank priors with data-driven deep priors. The PnP methods provide a partial solution to the interpretability limitations of E2E methods, yet they continue to grapple with the inherent challenge of slow reconstruction speeds characteristic of traditional model-based approaches.

(3) The DUN methods usually utilize multi-stage networks instead of the iterative optimization processes seen in model-based methods. Each stage employs deep neural network models to address both the data and prior subproblems iteratively. For instance, GAP-Net [34] transforms the generalized alternating projection optimization algorithm into a multi-stage network tailored for HSI reconstruction. ADMM-Net [35] unfolds the standard tensor alternating direction method of multipliers optimization algorithm into a multi-layer network structure. DSSP [36] enhances the spatial–spectral fidelity of HSI data by leveraging the local coherence and dynamic features of HSI to construct prior learning. DNU [37] creates a regularization term by exploring both local and non-local correlations of HSI as data-driven priors. Zhang et al. [38] proposes developing a deep Canonical Polyadic decomposition model in the unfolded network to learn a low-rank prior for HSI data. DGSMP [39] combines a learnable Gaussian Scale Mixture prior with the Maximum A Posteriori estimation algorithm to achieve HSI reconstruction. DGSM-Swin [40], a variant of DGSM [39], is constructed by leveraging the Swin Transformer [62], further enhancing the reconstruction performance of HSI. GAP-CCot [41] combines the expressive capabilities of convolution and the content Transformer, creating a hybrid network module that is integrated into the GAP algorithm for SCI reconstruction. HerosNet [42] unfolds the Iterative Shrinkage-Thresholding Algorithm (ISTA) [63] into a multi-stage network, inserting learnable flexible sensing matrices and constructing adaptive dynamic gradient descent at each stage. Ying et al. [43] employ a dual-domain feature learning approach to comprehensively acquire complementary information in the feature space, enhancing the overall algorithm's feature modeling capability. DAUHST [44] introduces an unfolding network framework that is aware of degradation, where parameters estimated during the degradation process govern different iterative stages. PADUT [45] designs a framework in-

corporating pixel-adaptive steps and a non-local spectral Transformer to enhance expressive capability at each stage. RDLUF [46] introduces a strategy for residual degradation learning that links the sensing matrix with the degradation process. This integration enhances spectral–spatial representation capabilities by incorporating both spectral and spatial priors. D^{2}PL-Net [47] endeavors to dynamically learn the actual degradation matrix throughout the deep unfolding network process to enhance HSI reconstruction, thereby narrowing the disparity between ideal and real-world degradations. DADF-Net [48] integrates the underlying connection between the network input and the true HSI, introducing a dynamic Fourier network to achieve high-quality HSI reconstruction. EDUNet [49] innovatively introduces a memory-assisted descent method based on momentum acceleration and a cross-stage spectral self-attention network to model the gradient-driven update module and the proximal mapping module, respectively. DPU [50] implements an HSI reconstruction model based on dual prior unfolding, which improves iteration efficiency by jointly utilizing multiple deep priors while strategically incorporating focused attention into the framework to enhance reconstruction quality. The DUN methods combine interpretability with high-quality reconstruction. Nonetheless, the current methods can still be enhanced by improving feature expression and addressing the challenge of cross-stage information loss.

3. Method

3.1. Preliminary

The complete physical imaging process for the Single-Disperser Coded Aperture Snapshot Spectral Imaging (SD-CASSI) system for hyperspectral image (HSI) data is shown in Figure 1. The SD-CASSI system is composed of a set of physical optical devices with different functions, designed to compress 3D HSI data into 2D compressed measurements. In detail, as the objective lens acquires 3D hyperspectral data, each spectral band undergoes modulation along the spatial dimension by the coded aperture. The modulated data subsequently pass through an optical dispersion element to shift each spectral band. Ultimately, all the spectral bands after dispersion are superimposed along the spectral direction to obtain the processed 2D compression measurement data. The 3D HSI data are denoted as $\mathcal{X} \in \mathbb{R}^{H \times W \times N_\lambda}$, where H, W and N_λ represent the height, width and spectral bands of 3D HSI data, respectively. The matrix $M \in \mathbb{R}^{H \times W}$ stands for the coded aperture. Hence, the physical imaging process of the 3D HSI can be described as follows:

$$\mathcal{X}'(:,:,n_\lambda) = \mathcal{X}(:,:,n_\lambda) \odot M \tag{1}$$

where $\mathcal{X}' \in \mathbb{R}^{H \times W \times N_\lambda}$ denotes the modulated cube, $n_\lambda \in [1, \ldots, N_\lambda]$ indexes the spectral bands, and $\odot$ denotes the element-wise product. The modulated HSI data, after being spatially shifted and summed element-wise across different spectral bands, can be represented as follows:

$$Y(m,n) = \sum_{n_\lambda=1}^{N_\lambda} \mathcal{X}'(m, n + d_\lambda, n_\lambda) + N \tag{2}$$

where $Y \in \mathbb{R}^{H \times (W + N_\lambda - 1)}$ represents the 2D compressive measurement; m and n represent the spatial coordinates; and d_λ represents the shifting distance. $N \in \mathbb{R}^{H \times (W + N_\lambda - 1)}$ represents the noise. In summary, the vectorized representation of the SD-CASSI system for 3D HSI data is described as follows:

$$\boldsymbol{y} = \Phi \boldsymbol{x} + \boldsymbol{n} \tag{3}$$

where $\boldsymbol{x} \in \mathbb{R}^{HWN_\lambda}$, $\boldsymbol{y} \in \mathbb{R}^{H(W+N_\lambda-1)}$, $\boldsymbol{n} \in \mathbb{R}^{H(W+N_\lambda-1)}$ denote the vectorized form of $\mathcal{X}$, Y, and N, respectively. $\Phi \in \mathbb{R}^{H(W+N_\lambda-1) \times HWN_\lambda}$ denotes the sensing matrix. After modeling the system's imaging model, the primary challenge of the HSI snapshot compressive imaging task is to reconstruct the original 3D HSI scene $\boldsymbol{x}$ given $\boldsymbol{y}$ and Φ. In the following sections, we will separately introduce the overall framework of the algorithm, the sparse spatial–spectral Transformer module, the wavelet-based intra-stage fusion module, and the wavelet-based inter-stage fusion module.

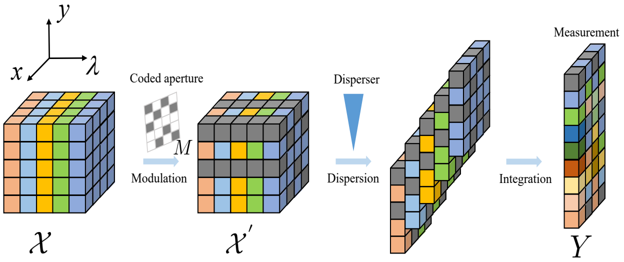


Figure 1. The physical imaging process of the SD-CASSI system for HSI data.

3.2. Overall Algorithm Framework

The overall imaging and reconstruction process is shown in Figure 2a. For the given raw HSI datum $\boldsymbol{x}$, the observed datum $\boldsymbol{y}$ is obtained after passing through the CASSI imaging system. Subsequently, the overall reconstruction process can be modeled as the following optimization problem:

$$\boldsymbol{x} = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{y} - \Phi\boldsymbol{x}\|_2^2 + \lambda\psi(\boldsymbol{x}) \tag{4}$$

where $\frac{1}{2}\|\boldsymbol{y} - \Phi\boldsymbol{x}\|_2^2$ represents the data fidelity term in the optimization problem, $\psi(\boldsymbol{x})$ represents the prior term, and λ represents the weighting parameter. Subsequently, the proximal gradient descent algorithm is employed to solve this ill-posed optimization inverse problem. Specifically, the overall optimization problem is transformed into a gradient descent operation and a proximal mapping operation. The solution process is as follows:

$$\boldsymbol{r}^{(k)} = \boldsymbol{x}^{(k-1)} - \rho\Phi^{\top}(\Phi\boldsymbol{x}^{(k-1)} - \boldsymbol{y}) \tag{5}$$

$$\boldsymbol{x}^{(k)} = \mathrm{prox}_{\lambda,\psi}(\boldsymbol{r}^{(k)}) \tag{6}$$

where Equation (5) represents the gradient descent operation, and Equation (6) represents the proximal mapping operation. Here, $\boldsymbol{x}^{(k-1)}$ and $\boldsymbol{x}^{(k)}$ denote the reconstruction results at iteration $\boldsymbol{k}-1$ and $\boldsymbol{k}$, respectively. The variable ρ represents the step size, and $\boldsymbol{r}^{(k)}$ represents the intermediate variable.

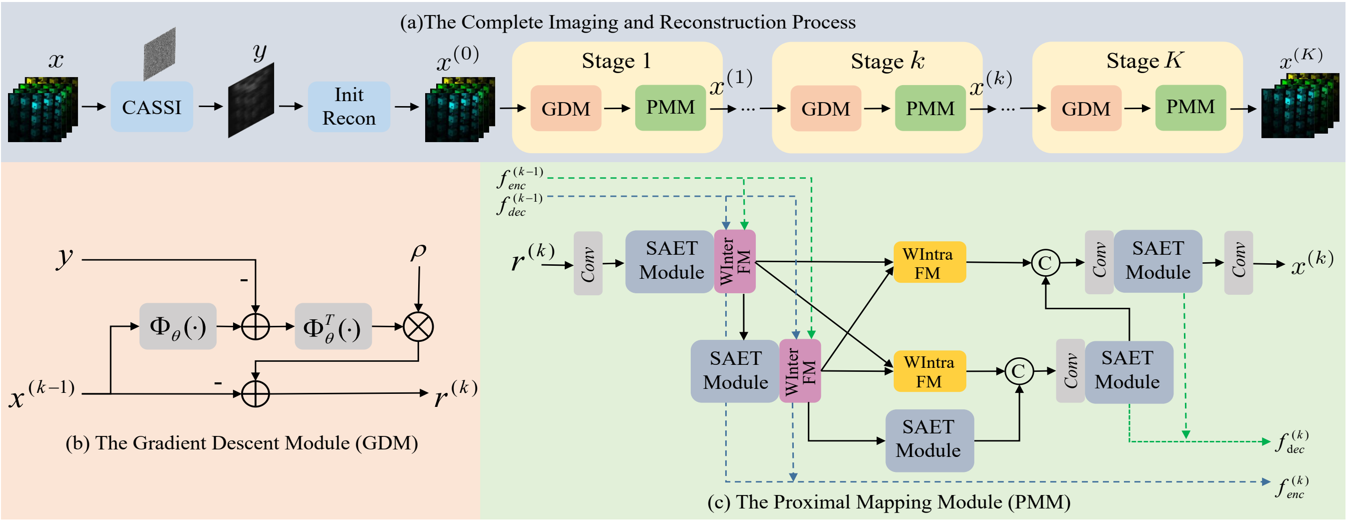
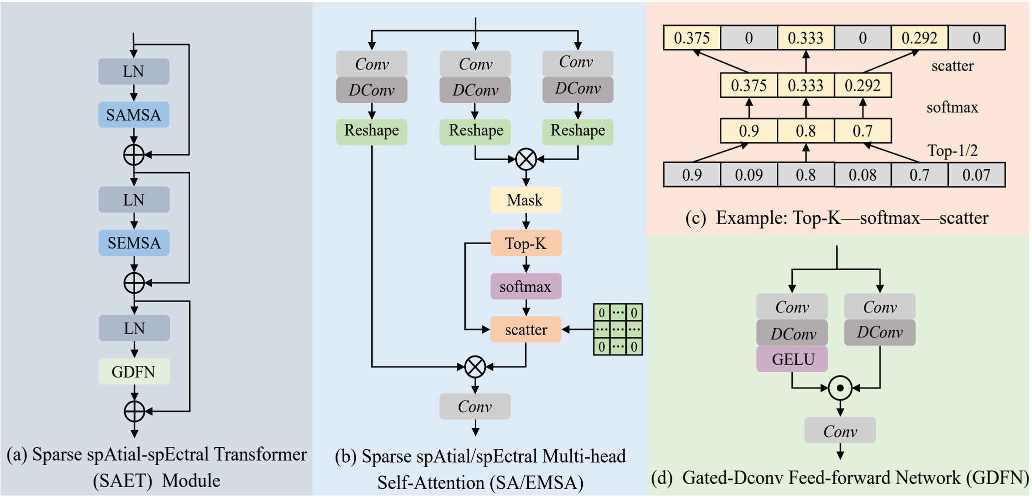


Figure 2. Overall algorithm framework. (**a**) The complete imaging and reconstruction process of the system. (**b**) The network structure of the gradient descent module. (**c**) The network structure of the proximal mapping module.

Through the iterative optimization process of $\boldsymbol{x}^{(k)}$ and $\boldsymbol{r}^{(k)}$, the final x can be reconstructed. Subsequently, we unfold the different iterative processes into different stages of deep unfolding network architecture, introducing gradient descent modules and proximal mapping modules to replace traditional operations. As shown in Figure 2a, each stage of the unfolded network contains a gradient descent module and a proximal mapping module. The structure of the gradient descent module is shown in Figure 2b. To enhance the representative capacity of the overall network framework, we construct $\Phi_\theta(\cdot)$ and $\Phi_\theta^\top(\cdot)$ by adding residual networks after the operations of matrices Φ and $\Phi^\top$. The structure of the proximal mapping module is shown in Figure 2c. We adopt a conventional U-Net structure, constructing a sparse spatial–spectral Transformer (SAET) module in the encoder–decoder parts of each layer to enhance the feature representation capabilities. Additionally, we introduce the Wavelet-based Intra-stage Fusion Module (WIntraFM) and Wavelet-based Inter-stage Fusion Module (WInterFM) to reduce feature loss within and across stages. The detailed structures of the different modules are described in the following sections.

3.3. The Sparse Spatial–Spectral Transformer (SAET) Module

The overview of the sparse spatial–spectral Transformer module is shown in Figure 3a. LN represents the Layer Normalization layer, SAMSA and SEMSA represent Sparse Spatial Multi-Head Self-Attention and Sparse Spectral Multi-Head Self-Attention, respectively, and GDFN represents the Gated-Dconv Feed-forward Network. We compose the final SAET module by serially connecting different network parts and utilizing skip connections.



(a) Sparse spAtial-spEctral Transformer (SAET) Module

(b) Sparse spAtial/spEctral Multi-head Self-Attention (SA/EMSA)

(c) Example: Top-K—softmax—scatter

(d) Gated-Dconv Feed-forward Network (GDFN)

Figure 3. (**a**) The structure of the sparse spatial–spectral Transformer module. (**b**) The structure of the sparse spatial/spectral multi-head self-attention. (**c**) A specific example of a correlation map sparsification. (**d**) The detailed structure of the gated-dconv feed-forward network.

The structures of SAMSA and SEMSA are shown in Figure 3b. Both SAMSA and SEMSA use the same method to transform the input features into queries, keys, and values. The difference lies in the way they compute the self-attention maps. Specifically, for a given input feature $X_{in} \in \mathbb{R}^{H \times W \times C}$, SA/EMSA employs a convolution layer *Conv* with 1×1 kernels and a depth-wise convolution layer *DConv* with 3×3 kernels to embed X_{in} and generate $Q \in \mathbb{R}^{H \times W \times C}$, $K \in \mathbb{R}^{H \times W \times C}$, and $V \in \mathbb{R}^{H \times W \times C}$. The formulas are expressed as follows:

$$Q = W_d^Q W_p^Q X_{in}, K = W_d^K W_p^K X_{in}, V = W_d^V W_p^V X_{in} \tag{7}$$

where $W_p^{(\cdot)}$ and $W_q^{(\cdot)}$ represent the weight parameters of the *Conv* layer and the *DConv* layer, respectively.

For SAMSA, the input features are split along the spatial dimension into non-overlapping windows of size $M \times M$. Each pixel within a window is treated as a token, and self-attention maps are computed within each window. Therefore, the query Q, key K, and value V features can be reshaped into $Q_A, K_A, V_A \in \mathbb{R}^{\frac{HW}{M^2} \times M^2 \times C}$. Subsequently, the Q_A, K_A, and V_A features are split along the last dimension into h heads and represented as: $Q_A = [Q_A^1, \cdots, Q_A^h]$, $K_A = [K_A^1, \cdots, K_A^h]$, $V_A = [V_A^1, \cdots, V_A^h]$. The dimension of each head is $d\frac{h}{A} = \frac{C}{h}$. The query Q_A^i and key K_A^i are dot-multiplied to obtain a self-attention map *Mask* of size $\frac{HW}{M^2} \times M^2 \times M^2$, representing the correlation of all pixels within each window. However, in practice, not all pixels are correlated. By applying sparsification to extract only the relevant pixels for correlation computation, the expressiveness of the self-attention mechanism can be further enhanced. Specifically, for the original self-attention map *Mask*, we use a 'Top-k' operation to select the top-k pixels with the highest correlation. Then, we apply 'Softmax' normalization to compute correlation coefficients for the selected pixels. Finally, we use a 'scatter' operation to return the correlation coefficients to the corresponding positions in the original self-attention map *Mask*, replacing positions where no correlation coefficient exists with zeros. SAMSA can be formally expressed as follows:

$$Mask = Q_A^i (K_A^i)^\top \tag{8}$$

$$\mathrm{head_i} = \mathrm{SparseAtten}(Q_A^i, K_A^i, V_A^i) \tag{9}$$

$$= \mathrm{scatter}(\mathrm{Softmax}(\frac{\text{Top-k}(Mask)}{\sqrt{d_A^h}}))V_A^i \tag{10}$$

$$X_{out} = W_p \mathrm{Concat}(\mathrm{head_1}, \cdots, \mathrm{head_n}) + X_{in} \tag{11}$$

where 'SparseAtten' represents the sparse self-attention operation, 'Concat' represents the feature concatenation operation, W_p represents the convolutional layer operation, and $X_{out} \in \mathbb{R}^{H \times W \times C}$ represents the feature output. To better understand the sparsification operation, we present an example of the 'Top-k' to 'Softmax' to 'scatter' operations as shown in Figure 3c.

For SEMSA, the spatial dimensions of the input features are transformed into column vectors of size HW. Each feature channel is treated as a token, and the self-attention map is obtained by calculating the correlations among all channels. Therefore, the query Q, key K, and value V features can be reshaped into $Q_E, K_E, V_E \in \mathbb{R}^{HW \times C}$. Subsequently, we split Q_E, K_E and V_E features along the spectral channel dimension into h heads, represented as $Q_E = [Q_E^1, \cdots, Q_E^h]$, $K_E = [K_E^1, \cdots, K_E^h]$, $V_E = [V_E^1, \cdots, V_E^h]$. The dimension of each head is $d\frac{h}{E} = \frac{C}{h}$. Next, the self-attention map *Mask* of size $\frac{C}{h} \times \frac{C}{h}$ for each head can be obtained by the dot product of Q_E^i and K_E^i, representing the correlations among all channels in the spectral dimension. After obtaining the self-attention map *Mask*, we apply the same sparsification operation as in SAMSA to extract the correlated channels and compute the correlation coefficients. Finally, the formal expression of the SEMSA block is as follows:

$$Mask = (K_E^i)^\top Q_E^i \tag{12}$$

$$\mathrm{head_i} = \mathrm{SparseAtten}(Q_E^i, K_E^i, V_E^i) \tag{13}$$

$$= \mathrm{scatter}(\mathrm{Softmax}(\frac{\text{Top-k}(Mask)}{\sqrt{d_E^h}}))V_E^i \tag{14}$$

$$X_{out} = W_p \mathrm{Concat}(\mathrm{head_1}, \cdots, \mathrm{head_n}) + X_{in} \tag{15}$$

3.4. The Wavelet-Based Intra-Stage Fusion Module (WIntraFM)

Since the proximal mapping network adopts a U-Net structure, there are up-sampling and down-sampling operations in different encoder–decoder features, leading to feature information loss. Discrete wavelet transform inherently possesses scale down-sampling characteristics, so we construct a wavelet-based intra-stage feature fusion module. The specific structure is shown in Figure 4.

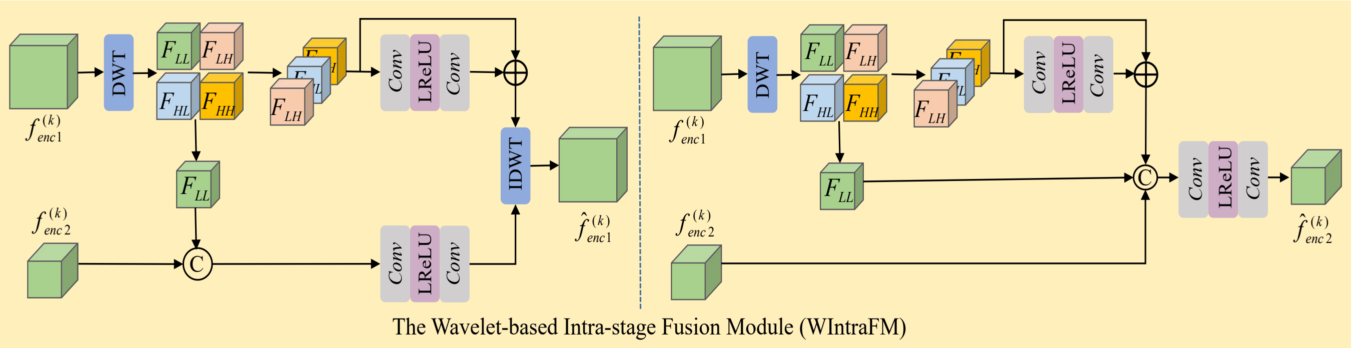


Figure 4. The wavelet-based intra-stage fusion module.

For two encoder output features $f^{(k)}_{enc1}$ and $f^{(k)}_{enc2}$ at different spatial scales, we apply the wavelet transform operation only to the large-scale feature $f^{(k)}_{enc1}$ to leverage the multi-scale characteristics of the wavelet transform. The expression is as follows:

$$F^{LL}_{enc1}, F^{LH}_{enc1}, F^{HL}_{enc1}, F^{HH}_{enc1}, = \mathrm{DWT}(f^{(k)}_{\mathrm{enc}1}) \tag{16}$$

where 'DWT' represents the discrete wavelet transform. In this paper, we specifically use the Haar wavelet transform. F^{LL}_{enc1} denotes the low-frequency component of the feature $f^{(k)}_{enc1}$, while $[F^{LH}_{enc1}, F^{HL}_{enc1}, F^{HH}_{enc1}]$ represents the high-frequency components of the feature $f^{(k)}_{enc1}$. Due to the reduced scale of the features after the wavelet transform, the spatial resolution of the feature $f^{(k)}_{enc2}$ matches that of the features $[F^{LL}_{enc1}, F^{LH}_{enc1}, F^{HL}_{enc1}, F^{HH}_{enc1}]$. Subsequently, we perform feature fusion on features of the same size to reduce information loss.

To obtain the high-resolution encoder feature after fusion, we concatenate the low-frequency component F^{LL}_{enc1} with feature $f^{(k)}_{enc2}$ and perform a low-frequency enhancement operation to obtain the fused low-frequency information. Additionally, we enhance the high-frequency components $[F^{LH}_{enc1}, F^{HL}_{enc1}, F^{HH}_{enc1}]$ separately to obtain the fused high-frequency information. Finally, we perform an inverse wavelet transform on the fused low-frequency and high-frequency features to obtain the final fused feature. The formal expression is as follows:

$$\hat{F}_L = \mathrm{Conv}_{3\times3}(\mathrm{LRelu}(\mathrm{Conv}_{1\times1}(\mathrm{Concat}(f^{(k)}_{enc2}, F^{LL}_{enc1})))) \tag{17}$$

$$\hat{F}_H = [F^{LH}_{enc1}, F^{HL}_{enc1}, F^{HH}_{enc1}] + \mathrm{Conv}_{1\times1}(\mathrm{LRelu}(\mathrm{Conv}_{1\times1}([F^{LH}_{enc1}, F^{HL}_{enc1}, F^{HH}_{enc1}]))) \tag{18}$$

$$\hat{f}^{(k)}_{enc1} = \mathrm{IDWT}(\hat{F}_L, \hat{F}_H) \tag{19}$$

where 'Conv$_{1\times1}$' and 'Conv$_{3\times3}$' represent convolutional layers with kernel sizes of 1 × 1 and 3 × 3, respectively. 'Concat' represents the feature concatenation operation, 'LRelu' represents the activation function, and 'IDWT' represents the inverse discrete wavelet transform. The variables $\hat{F}_L$ and $\hat{F}_H$ represent the fused low-frequency and high-frequency features, respectively, and $\hat{f}^{(k)}_{enc1}$ represents the output high-resolution fused feature.

To obtain the low-resolution encoder feature after fusion, we first enhance the high-frequency components $[F^{LH}_{enc1}, F^{HL}_{enc1}, F^{HH}_{enc1}]$ separately to get the fused high-frequency component $\hat{F}'_H$. Next, we concatenate the fused $\hat{F}'_H$ with the low-frequency component F^{LL}_{enc1} and feature $f^{(k)}_{enc2}$, and further enhance these features to obtain the final low-resolution fused feature $\hat{f}^{(k)}_{enc2}$. The formal expression is as follows:

$$\hat{F}'_H = [F^{LH}_{enc1}, F^{HL}_{enc1}, F^{HH}_{enc1}] + \mathrm{Conv}_{1\times1}(\mathrm{LRelu}(\mathrm{Conv}_{1\times1}([F^{LH}_{enc1}, F^{HL}_{enc1}, F^{HH}_{enc1}]))) \tag{20}$$

$$\hat{f}^{(k)}_{enc2} = \mathrm{Conv}_{3\times3}(\mathrm{LRelu}(\mathrm{Conv}_{1\times1}(\mathrm{Concat}(\hat{F}'_H, F^{LL}_{enc1}, f^{(k)}_{enc2})))) \tag{21}$$

3.5. The Wavelet-Based Inter-Stage Fusion Module (WInterFM)

In the deep unfolding network, the continuous transformation of signals to features across different stages leads to feature information loss. Wavelet transform can effectively learn and emphasize high-frequency details in features. Therefore, we design a wavelet-based inter-stage feature fusion module to mitigate the information loss across stages. The specific structure is shown in Figure 5.

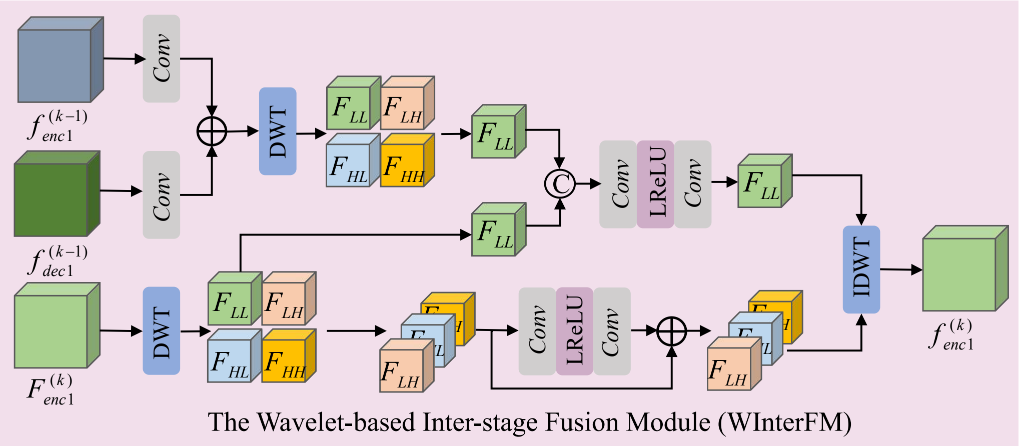


Figure 5. The wavelet-based inter-stage fusion module.

Specifically, taking the inter-stage fusion operation of the high-resolution encoder feature $F_{enc1}^{(k)}$ at the k stage as an example, we first add the encoder feature $f_{enc1}^{(k-1)}$ and decoder feature $f_{dec1}^{(k-1)}$ from the $k-1$ stage to obtain feature $F^{(k-1)}$. Then, we perform a discrete wavelet transform on feature $F^{(k-1)}$ to obtain the high-frequency components $[F_{LH}^{(k-1)}, F_{HL}^{(k-1)}, F_{HH}^{(k-1)}]$ and low-frequency component $F_{LL}^{(k-1)}$. Additionally, we perform a wavelet transform on the feature $F_{enc1}^{(k)}$ from the k stage to obtain the high-frequency components $[F_{LH}^{(k)}, F_{HL}^{(k)}, F_{HH}^{(k)}]$ and low-frequency component $F_{LL}^{(k)}$. Next, we concatenate the low-frequency components $F_{LL}^{(k)}$ and $F_{LL}^{(k-1)}$, followed by a low-frequency enhancement to obtain the low-frequency component $f_{LL}^{(k)}$ of the inter-stage fusion feature. To reduce noise impact, we only perform high-frequency feature enhancement on the high-frequency components $[F_{LH}^{(k)}, F_{HL}^{(k)}, F_{HH}^{(k)}]$ from stage k to obtain the fused high-frequency components $[f_{LH}^{(k)}, f_{HL}^{(k)}, f_{HH}^{(k)}]$. Finally, we perform an inverse wavelet transform on the high-frequency components $[f_{LH}^{(k)}, f_{HL}^{(k)}, f_{HH}^{(k)}]$ and low-frequency component $f_{LL}^{(k)}$ to obtain the final inter-stage fusion feature $f_{enc1}^{(k)}$. The formal expressions are as follows:

$$F^{(k-1)} = \mathrm{Conv}_{3\times3}(f_{enc1}^{(k-1)}) + \mathrm{Conv}_{3\times3}(f_{dec1}^{(k-1)}) \tag{22}$$

$$F_{LL}^{(k-1)},\ F_{LH}^{(k-1)}, F_{HL}^{(k-1)}, F_{HH}^{(k-1)} = \mathrm{DWT}(F^{(k-1)}) \tag{23}$$

$$F_{LL}^{(k)}, F_{LH}^{(k)}, F_{HL}^{(k)}, F_{HH}^{(k)} = \mathrm{DWT}(F_{enc1}^{(k)}) \tag{24}$$

$$f_{LL}^{(k)} = \mathrm{Conv}_{3\times3}(\mathrm{LRelu}(\mathrm{Conv}_{1\times1}(\mathrm{Concat}(F_{LL}^{(k)}, F_{LL}^{(k-1)})))) \tag{25}$$

$$[f_{LH}^{(k)}, f_{HL}^{(k)}, f_{HH}^{(k)}] = \mathrm{Conv}_{1\times1}(\mathrm{LRelu}(\mathrm{Conv}_{1\times1}([F_{LH}^{(k)}, F_{HL}^{(k)}, F_{HH}^{(k)}]))) + [F_{LH}^{(k)}, F_{HL}^{(k)}, F_{HH}^{(k)}] \tag{26}$$

$$f_{enc1}^{(k)} = \mathrm{IDWT}(f_{LL}^{(k)}, [f_{LH}^{(k)}, f_{HL}^{(k)}, f_{HH}^{(k)}]) \tag{27}$$

where ‘$\mathrm{Conv}_{1\times1}$’ and ‘$\mathrm{Conv}_{3\times3}$’ represent convolutional layers with kernel sizes of 1 × 1 and 3 × 3, respectively. ‘Concat’ represents the feature concatenation operation, ‘LRelu’ represents the activation function, and ‘DWT’ and ‘IDWT’ represent the discrete wavelet transform and inverse transform.

4. Experimental Results

In this section, we conduct experiments on various datasets to validate the superiority of our method compared to other state-of-the-art methods. The specific content includes experimental setup, simulation results, real data results, and ablation study, among other subsections.

4.1. Experiment Setup

Simulated experiments on CAVE and KAIST datasets: The CAVE dataset [64] contains 32 HSI samples, each with spatial dimensions of 512 × 512. The KAIST dataset [65] consists of 30 HSI samples, each with spatial dimensions of 2704 × 3376. Both datasets feature 31 spectral bands, spanning wavelengths from 400 nm to 700 nm with 10 nm intervals. To ensure consistency with the experimental setups of other state-of-the-art methods, we also employ spectral interpolation to adjust each HSI sample to 28 spectral bands within the wavelength range of 450 nm to 650 nm. In the actual training and testing process, we maintain the same experimental settings as other state-of-the-art methods. The CAVE dataset is randomly cropped into HSI patches consistent with the size of the coded aperture to construct the training set. The coded aperture size is 256 × 256, and all training HSI patches are of size 256 × 256 × 28. Meanwhile, we select HSI patches of size 256 × 256 × 28 from 10 scenes in the KAIST dataset as the test set. At this point, the spatial receptive field of our proposed network is 256 × 256.

Simulated experiments on ARAD_1K dataset: The ARAD_1K dataset [66] provides a large-scale natural hyperspectral image dataset. Each HSI sample in the ARAD_1K dataset consists of 31 spectral bands, with a spatial resolution of 482 × 512, and covers a wavelength range of 400–700 nm. It contains 1000 hyperspectral images, of which 900 are used as the training set and 50 as the test set. The size of the coded aperture is set to 256 × 256, and each HSI block in the ARAD_1K training set is cropped to a size of 256 × 256 × 31. Similar to the KAIST dataset, our ARAD_1K test dataset is also constructed by cropping HSI patches of size 256 × 256 × 31 from 10 scenes.

Real-world scene dataset: For the real-world dataset, we use the 2D measurements from five real-scene samples provided by TSA-Net [24] to assess the effectiveness of our proposed method. Each 2D measurement has dimensions of 660 × 714, and the coded aperture is sized at 660 × 660. To build the training dataset, we randomly crop the CAVE and KAIST datasets into data blocks of size 660 × 660 × 28.

Experimental settings: We use Root Mean Square Error (RMSE) as the loss function for the algorithm. Additionally, all related experiments are constructed based on the PyTorch deep learning framework and conducted on an NVIDIA RTX 3090 GPU. The algorithm is trained using the Adam optimizer with an initial learning rate set to 0.0001 and a maximum of 200 training epochs. The spectral shift step for all HSI data is configured to 2. We also use the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity (SSIM) Index [67] as metrics to evaluate the reconstruction performance of different algorithms. PSNR and SSIM serve as two standards for spatial quality assessment, measuring visual quality and structural similarity, respectively. Higher values for both PSNR and SSIM indicate improved spatial reconstruction.

4.2. Simulation Results on CAVE and KAIST

The objective results comparison for the KAIST dataset test scenes is shown in Table 1. We select nine state-of-the-art methods as comparison methods, including four E2E methods (TSA-Net [24], HDNet [26], MST [25], and CST [27]) and five DUN methods (DGSMP [39], HerosNet [42], DAUHST [44], PADUT [45], and RDLUF [46]). We present the PSNR and SSIM results for all comparison methods, with the best and second-best reconstruction results for all test scenes highlighted in red and blue, respectively. In addition, we calculate the number of parameters (Params) and floating-point operations (FLOPs) to evaluate the model complexity of all comparison methods. As seen in Table 1, our method achieves the best average reconstruction results across all test scenes and delivers the best reconstruction results in most of the individual test scenes. Meanwhile, our method also shows a certain advantage in complexity compared to other methods. Our method achieves a PSNR of 39.76 dB and an SSIM of 0.979. Specifically, compared to the current leading E2E method, CST [27], our results demonstrate an improvement of 3.91 dB in PSNR and 0.0025 in SSIM. Furthermore, compared to the top DUN method, RDLUF [46], our approach improves PSNR by 0.31 dB and SSIM by 0.002. Therefore, these objective results validate the effectiveness of our method on the KAIST test dataset.

Table 1. Comparison of results using the KAIST test scenes, with the PSNR metric (in dB) presented as the upper entry and the SSIM metric as the lower entry in each cell. The best results are highlighted in red, and the second-best results are highlighted in blue.

Algorithms	Params	FLOPs	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	Average
TSA-Net [24]	44.25 M	110.06 G	32.03 0.892	31.00 0.858	32.25 0.915	39.19 0.953	29.39 0.884	31.44 0.908	30.32 0.878	29.35 0.888	30.01 0.890	29.59 0.874	31.46 0.894
HDNet [26]	2.37 M	154.76 G	35.14 0.935	35.67 0.940	36.03 0.943	42.30 0.969	32.69 0.946	34.46 0.952	33.67 0.926	32.48 0.941	34.89 0.942	32.38 0.937	34.97 0.943
MST-L [25]	2.03 M	28.15 G	35.40 0.941	35.87 0.944	36.51 0.953	42.27 0.973	32.77 0.947	34.80 0.955	33.66 0.925	32.67 0.948	35.39 0.949	32.50 0.941	35.18 0.948
CST-L [27]	3.00 M	27.81 G	35.82 0.947	36.54 0.952	37.39 0.959	42.28 0.972	33.40 0.953	35.52 0.962	34.44 0.937	33.83 0.959	35.92 0.951	33.36 0.948	35.85 0.954
DGSMP [39]	3.76 M	646.65 G	33.26 0.915	32.09 0.898	33.06 0.925	40.54 0.964	28.86 0.882	33.08 0.937	30.74 0.886	31.55 0.923	31.66 0.911	31.44 0.925	32.63 0.917
HerosNet [42]	11.75 M	446.29 G	35.75 0.972	35.40 0.968	34.07 0.966	38.59 0.987	33.31 0.969	35.58 0.977	33.27 0.963	33.75 0.971	34.04 0.967	33.18 0.968	34.69 0.971
DAUHST [44]	6.15 M	79.50 G	37.25 0.958	39.02 0.967	41.05 0.971	46.15 0.983	35.80 0.969	37.08 0.970	37.57 0.963	35.10 0.966	40.02 0.970	34.59 0.956	38.36 0.967
PADUT [45]	5.38 M	90.46 G	37.36 0.962	40.43 0.978	42.38 0.979	46.62 0.990	36.26 0.974	37.27 0.974	37.83 0.966	35.33 0.974	40.86 0.978	34.55 0.963	38.89 0.974
RDLUF [46]	1.89 M	231.09 G	37.74 0.967	40.76 0.979	43.05 0.981	47.59 0.992	36.93 0.978	37.54 0.978	38.34 0.971	35.57 0.974	42.18 0.982	34.77 0.964	39.45 0.977
Ours	2.25 M	121.43 G	37.85 0.970	40.80 0.980	43.10 0.982	48.12 0.993	37.48 0.980	37.52 0.979	38.63 0.971	36.41 0.979	42.04 0.982	35.62 0.970	39.76 0.979

To illustrate the qualitative results on the KAIST test dataset, we present a subjective comparison of the reconstruction outcomes for the test scenes using different methods. To better illustrate the comparison between the training and test datasets, we present the RGB images of 10 test scenes from the KAIST dataset and 10 training scenes from the CAVE dataset, shown in Figure 6 and Figure 7, respectively. The subjective comparison of the reconstruction results for scene S10 in the KAIST dataset is shown in Figure 8. We select data from four different spectral bands for the reconstruction results and compare the subjective effects of all methods. We crop and enlarge the yellow box areas in each spectral band image to facilitate a clearer comparison of the reconstruction differences. From Figure 8, it is evident that our proposed method exhibits clearer texture details across different spectral bands compared to other methods. This further validates the effectiveness of our approach on the KAIST test dataset.

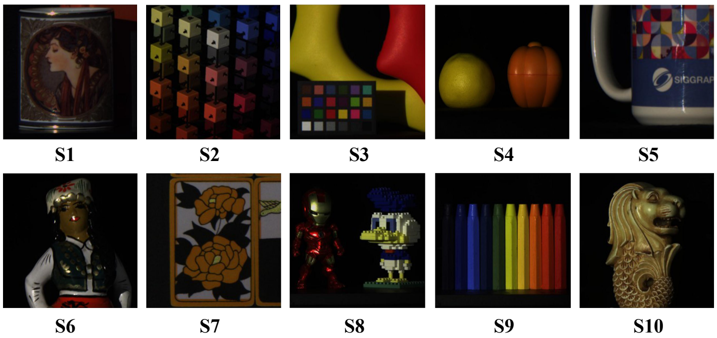


Figure 6. RGB images of ten test scenes from the KAIST dataset.

Figure 7. RGB images of 10 training scenes from the CAVE dataset.

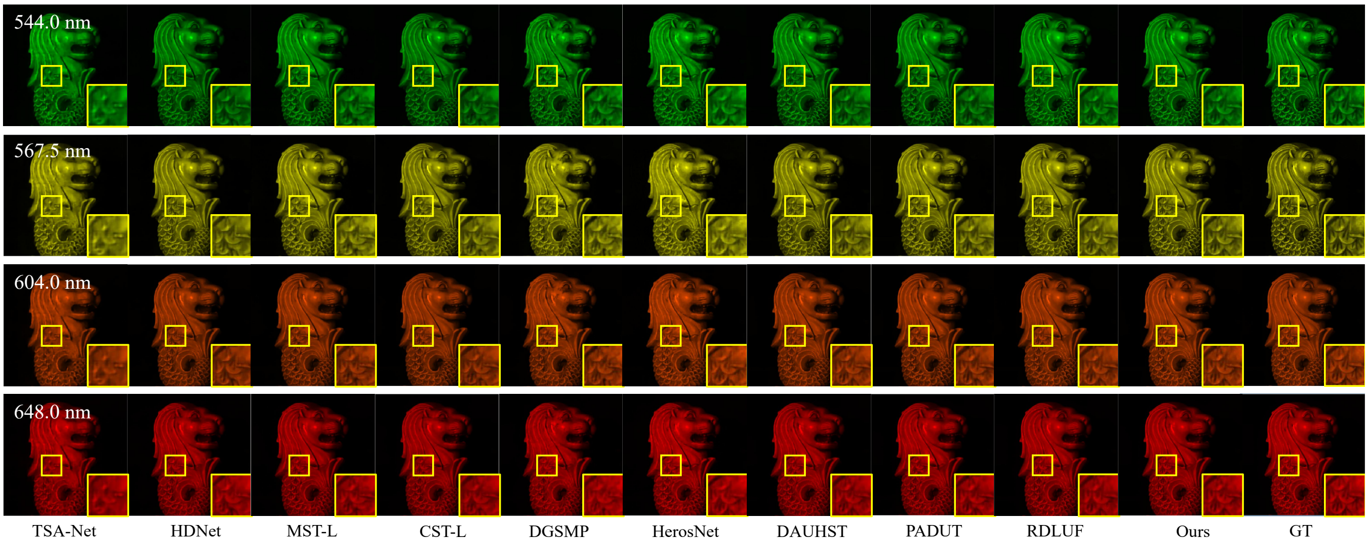


Figure 8. Comparison of the reconstruction results for scene S10 in the KAIST test dataset using different methods. The enlarged regions help to compare the reconstruction results better.

Additionally, to further compare the spectral consistency of reconstruction results from different methods, we select a small region from test scene S10 for evaluation as shown in Figure 9a. We present the spectral density curves of different reconstruction results and calculate the correlation coefficients between the reconstructed results and the ground truth data. Our method's reconstruction results are the closest to the ground truth spectral density curve and achieve the highest correlation coefficient, further validating the accuracy of our reconstruction.

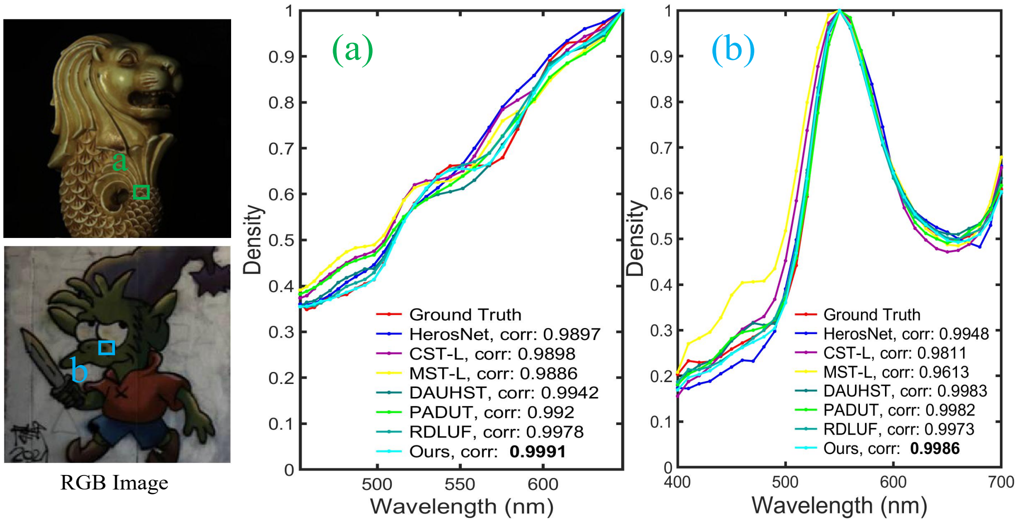


Figure 9. Spectral density curves of reconstruction results for KAIST test scene S10 and ARAD_1K test scene S5 using different methods. (**a**,**b**) The spectral density curves and correlation coefficients, respectively, for the selected regions.

4.3. Simulation Results on ARAD_1K

The objective results comparison for the ARAD_1K dataset test scenes is shown in Table 2. We select nine state-of-the-art methods as comparison methods, including four E2E methods (TSA-Net [24], HDNet [26], MST [25], CST [27]) and five DUN methods (DGSMP [39], HerosNet [42], DAUHST [44], PADUT [45], RDLUF [46]). We present the PSNR and SSIM results for all comparison methods, with the best and second-best reconstruction results for all test scenes highlighted in red and blue, respectively. In addition, we calculate the number of parameters (Params) and floating-point operations (FLOPs) to evaluate the model complexity of all comparison methods. As seen in Table 2, our method achieves the best average test results and optimal performance across all scenarios. Specifically, our method reached PSNR and SSIM metrics of 42.98 dB and 0.985, respectively. Compared to the next best method, our approach shows an average improvement of 0.2dB in PSNR and 0.001 in SSIM. These objective results validate the effectiveness of the proposed method on the ARAD_1K dataset.

To illustrate the qualitative results on the ARAD_1K test dataset, we present a subjective comparison of the reconstruction outcomes for the test scenes using different methods. The RGB images of all test scenes from the ARAD_1K dataset are shown in Figure 10. The subjective comparison of the reconstruction results for scene S5 is shown in Figure 11. We select reconstruction results from four different spectral bands to compare the subjective performance of all methods. To facilitate the comparison of differences in reconstruction results, we crop and enlarge the yellow box areas in each spectral band image. As shown in Figure 11, our method produces reconstruction results with fewer artifacts and clearer structural textures compared to other methods. This further validates the effectiveness of our approach on the ARAD_1K test dataset.

Additionally, to further compare the spectral consistency of reconstruction results from different methods, we also select a small region from test scene S5 for evaluation, as shown in Figure 9b. We present the spectral density curves of different reconstruction results and calculate the correlation coefficients between the reconstructed results and the ground truth data. Our method's reconstruction results are the closest to the ground truth spectral density curve and achieve the highest correlation coefficient, further validating the accuracy of our reconstruction.

Table 2. Comparison of results using the ARAD_1K test scenes, with the PSNR metric (in dB) presented as the upper entry and the SSIM metric as the lower entry in each cell. The best results are highlighted in red, and the second-best results are highlighted in blue.

Algorithms	Params	FLOPs	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	Average
TSA-Net [24]	44.25 M	110.06 G	33.79 0.950	28.38 0.877	27.11 0.886	33.36 0.924	25.85 0.831	25.09 0.671	29.49 0.888	20.88 0.687	21.34 0.782	31.88 0.875	27.72 0.837
HDNet [26]	2.37 M	154.76 G	35.18 0.935	28.96 0.862	28.54 0.875	35.54 0.918	26.58 0.816	31.40 0.823	30.22 0.874	29.99 0.793	29.77 0.884	32.93 0.885	30.91 0.866
MST-L [25]	2.03 M	28.15 G	37.89 0.962	31.44 0.903	31.06 0.917	36.81 0.935	29.79 0.897	35.05 0.906	32.83 0.921	32.62 0.858	34.09 0.941	34.99 0.923	33.66 0.916
CST-L [27]	3.00 M	27.81 G	41.06 0.978	34.09 0.938	33.40 0.948	39.25 0.957	32.18 0.935	38.78 0.955	35.16 0.953	34.62 0.901	37.06 0.972	36.72 0.947	36.23 0.948
DGSMP [39]	3.76M	646.65G	37.10 0.959	29.97 0.886	29.44 0.908	36.30 0.935	28.20 0.875	34.80 0.901	31.18 0.908	31.51 0.840	32.05 0.928	33.92 0.912	32.45 0.905
HerosNet [42]	11.75 M	446.29 G	38.17 0.981	33.27 0.958	32.11 0.955	39.31 0.982	29.44 0.929	33.90 0.951	32.72 0.948	32.22 0.926	33.18 0.977	33.96 0.947	33.83 0.955
DAUHST [44]	6.15 M	79.50 G	45.94 0.991	40.53 0.978	39.26 0.978	46.85 0.990	38.26 0.979	43.67 0.988	40.44 0.980	38.94 0.959	44.41 0.990	40.62 0.974	41.89 0.980
PADUT [45]	5.38 M	90.46 G	46.65 0.992	41.20 0.981	39.85 0.981	47.79 0.992	38.93 0.981	44.38 0.990	41.07 0.982	39.53 0.963	45.45 0.993	41.05 0.977	42.59 0.983
RDLUF [46]	1.89 M	231.09 G	46.50 0.992	41.15 0.980	40.35 0.982	48.16 0.993	38.77 0.981	44.61 0.991	41.09 0.982	40.21 0.970	45.98 0.993	41.01 0.976	42.78 0.984
Ours	2.25 M	121.43 G	46.72 0.993	41.47 0.982	40.41 0.982	48.18 0.993	39.12 0.983	44.73 0.992	41.16 0.983	40.53 0.970	46.29 0.995	41.17 0.977	42.98 0.985

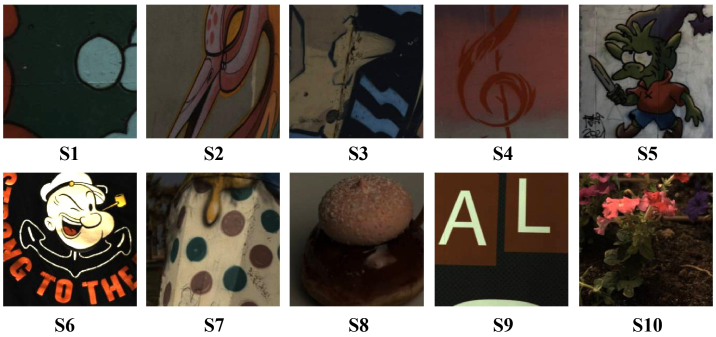


Figure 10. RGB images of ten test scenes from the ARAD_1K dataset.

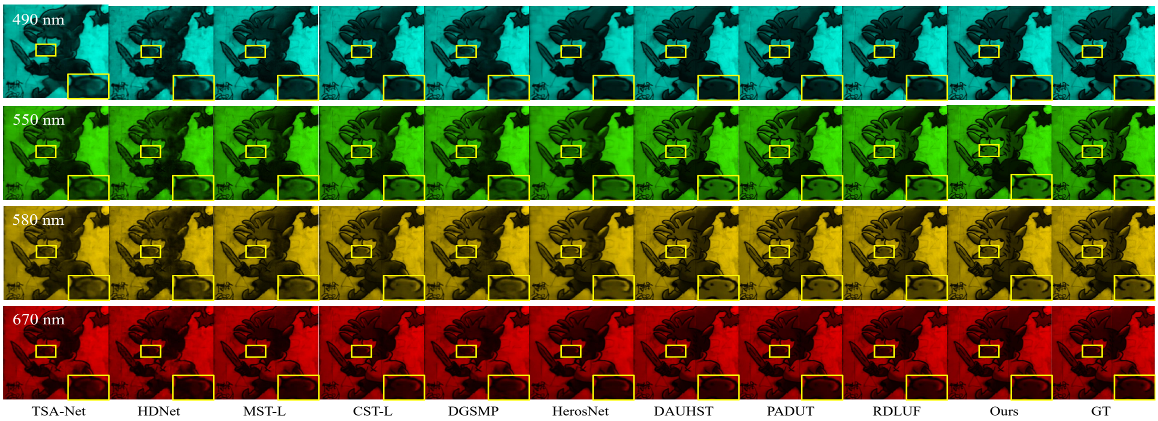


Figure 11. Comparison of the reconstruction results for scene S5 in the ARAD_1K test dataset using different methods. The enlarged regions help to compare the reconstruction results better.

4.4. Real Data Results

To validate the effectiveness of our method in real-world scenarios, we conduct tests on a real-world scene dataset. We compare our method with five advanced methods, including two E2E methods (TSA-Net [24], HDNet [26]) and three DUN methods (DGSMP [39], DAUHST [44], RDLUF [46]). Due to the large size of the test scenes, our method and other DUN methods are all compared using the same three-stage network. The RGB images of the five test scenes from the real-world scene dataset are shown in Figure 12. Since the dataset lacks ground truth data, we compare the reconstruction results of different methods by referencing the RGB images. We select scene S3 to present the subjective reconstruction results. As shown in Figure 13, our method produces smoother and clearer appearances in three different spectral bands. Notably, for the yellow-marked facial area, our reconstruction results retain structural content with fewer reconstruction artifacts. This demonstrates the effectiveness of our proposed method for real-world test data.

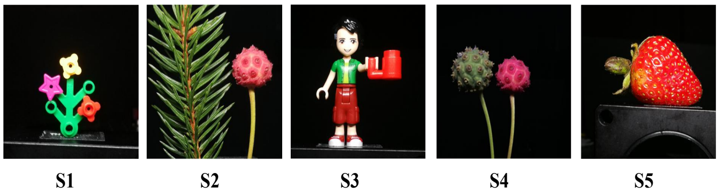


Figure 12. RGB images of five real-world test scenes.

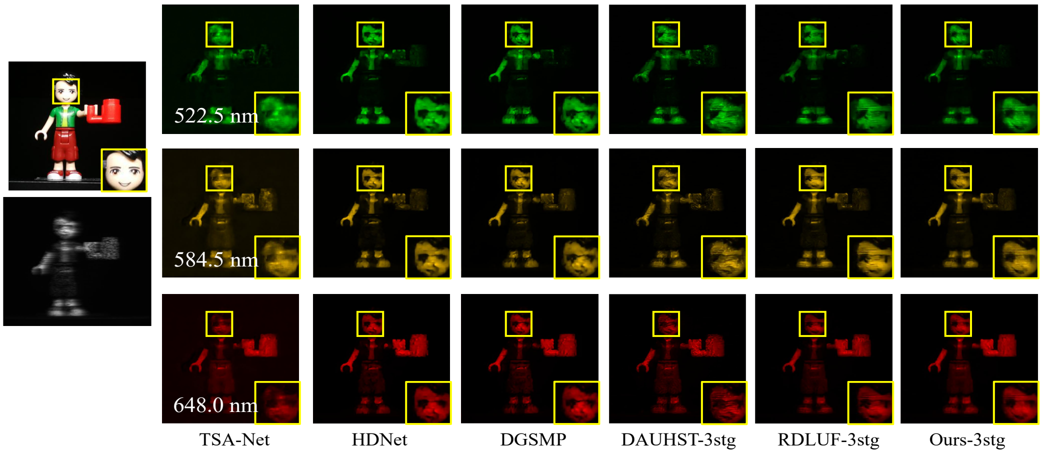


Figure 13. Subjective comparison of reconstruction results for three different spectral bands in real scene S3 using different methods. Enlarged areas are used to compare the differences in reconstruction results more clearly.

4.5. Ablation Study

To validate the impact of the number of stages on the overall performance of our algorithm, we conduct ablation experiments with different stage counts. As shown in Table 3, we evaluate the model complexity and average reconstruction performance of our algorithm with varying stage numbers on the CAVE and KAIST simulation datasets. Since our algorithm maintains parameter sharing within the network structure of each stage, the number of model parameters does not change with the number of stages. However, the number of floating-point operations (FLOPs) increases with the number of stages. The results indicate that the performance of our method improves with the increase in the number of stages. To balance model effectiveness and complexity, we ultimately adopt nine stages for the overall model configuration.

Table 3. The computational complexity and average reconstruction performance across various numbers of network stages on the KAIST test scenes.

Stage Number	Params (M)	FLOPs (G)	PSNR (dB)	SSIM
1	2.25	26.10	37.05	0.964
3	2.25	39.72	38.11	0.971
5	2.25	66.96	38.74	0.974
7	2.25	94.20	39.48	0.977
9	2.25	121.43	39.76	0.979

To verify the effectiveness of different modules in our method, we conduct ablation experiments with models composed of different structures. All experiments utilize models with nine stages and are tested on the CAVE and KAIST simulation datasets. Table 4 shows the average test reconstruction results and model complexity for different model configurations. We use the basic spatial–spectral Transformer structure as a baseline for comparison, as shown in row (a) of Table 4. To evaluate the effectiveness of different modules in enhancing reconstruction, we conduct four experiments: using only the SAET module, combining the SAET and WIntraF modules, combining the SAET and WInterF modules, and combining all modules. The results of these experiments are presented in rows (b) to (e) of Table 4, respectively. Specifically, from the results of row (b) in Table 4, it is evident that compared to the baseline model, the SAET model significantly enhances reconstruction performance, with improvements of 0.25 dB in PSNR and 0.001 dB in SSIM. This demonstrates the improvement in the overall model's expressive capability brought by our proposed sparse spatial–spectral Transformer network structure. Additionally, comparing rows (c) and (d) with row (b) in Table 4 shows that the model combining SAET and WIntraF achieves a 0.11 dB improvement in PSNR over the SAET-only model, while the model combining SAET and WInterF achieves a 0.14 dB improvement in PSNR over the SAET-only model. These results clearly validate the effectiveness of the wavelet-based intra-stage and inter-stage fusion modules. Finally, the reconstruction results of the complete model in row (e) verify the performance enhancement of the proposed model across various component structures.

Table 4. The effectiveness of different components.

Setting	SAET	WIntraF	WInterF	Params (M)	FLOPs (G)	PSNR (dB)	SSIM
(a) (Base)				1.85	109.51	39.33	0.977
(b)	√			1.85	109.51	39.58	0.978
(c)	√	√		1.90	112.34	39.69	0.978
(d)	√		√	2.21	118.60	39.72	0.978
(e) (Ours)	√	√	√	2.25	121.43	39.76	0.979

To validate the impact of different sparsity coefficients in the sparse spatial–spectral Transformer structure, we conduct ablation experiments with various sparsity coefficients. As shown in Table 5, the spatial Top-k and spectral Top-k rows represent the spatial and spectral sparsity coefficients, respectively. Different sparsity coefficients indicate the proportion of sparse elements in the correlation matrix relative to the total elements. When the spatial and spectral sparsity coefficients are both set to 1, all elements are selected as sparse elements, making the model structure identical to the original spatial–spectral Transformer structure. According to Table 5, the optimal reconstruction performance is achieved when the spatial sparsity coefficient is 1/3 and the spectral sparsity coefficient is 4/5, resulting in a 0.25 dB improvement in PSNR compared to the original spatial–spectral Transformer structure. Therefore, in the overall algorithm, we select 1/2 and 4/5 as the sparsity coefficients for the spatial and spectral dimensions, respectively.

Table 5. The impact of spatial and spectral sparsity coefficients in the SAET module, where Top-k represents the proportion of selected elements to all elements.

Spatial Top-k	1	1/2	1/2	1/2	1/3	1/4	2/3
Spectral Top-k	1	1	4/5	3/4	4/5	4/5	4/5
PSNR (dB)	39.33	39.53	39.52	39.45	39.58	39.32	39.45
SSIM	0.977	0.975	0.978	0.977	0.978	0.976	0.977

5. Conclusions

In this study, we propose a deep unfolding network that hybridizes sparse Transformer and wavelet fusion for the snapshot compressive imaging of hyperspectral images (HSIs). Firstly, since not all elements in hyperspectral images are correlated in the spatial and spectral dimensions, we introduce a spatial–spectral sparse Transformer technique to enhance the feature representation capability of the algorithm. Then, to address the issue of feature information loss due to scale transformation within stages, we propose a wavelet-based intra-stage feature fusion method. Finally, we introduce a wavelet-based inter-stage feature fusion method to tackle feature information loss caused by signal-to-feature conversions between stages. Experiments on various simulated and real datasets further validate that the proposed algorithm achieves superior hyperspectral image reconstruction results. However, the current method's utilization of wavelet transforms is still preliminary. In future work, we will explore how to learn the sparsity coefficients to better accommodate different model structures. Additionally, we will continue to explore how to utilize wavelet transform features better to design network structures, thereby further improving the algorithm's performance.

Author Contributions: Conceptualization, Y.Y. and J.W.; methodology, Y.Y.; software, Y.Y.; validation, Y.Y.; formal analysis, Y.Y. and J.W.; writing—original draft preparation, Y.Y.; writing—review and editing, J.W., Y.S. and N.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key R&D Program of China (No. 2021ZD0111902), NSFC (62272016, 62372018, U21B2038).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Pan, Z.; Healey, G.; Prasad, M.; Tromberg, B. Face recognition in hyperspectral images. *IEEE Trans. Pattern Anal. Mach. Intell.* **2003**, *25*, 1552–1560.
2. Huang, Y.; Peng, J.; Sun, W.; Chen, N.; Du, Q.; Ning, Y.; Su, H. Two-branch attention adversarial domain adaptation network for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–13. [CrossRef]
3. Zhu, L.; Wu, J.; Biao, W.; Liao, Y.; Gu, D. Spectralmae: Spectral masked autoencoder for hyperspectral remote sensing image reconstruction. *Sensors* **2023**, *23*, 3728. [CrossRef]
4. Wang, X.; Chen, J.; Wei, Q.; Richard, C. Hyperspectral image super-resolution via deep prior regularization with parameter estimation. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *32*, 1708–1723. [CrossRef]
5. Liu, S.; Li, Z.; Wang, G.; Qiu, X.; Liu, T.; Cao, J.; Zhang, D. Spectral–Spatial Feature Fusion for Hyperspectral Anomaly Detection. *Sensors* **2024**, *24*, 1652. [CrossRef]
6. He, C.; Wei, Y.; Guo, K.; Han, H. Removal of Mixed Noise in Hyperspectral Images Based on Subspace Representation and Nonlocal Low-Rank Tensor Decomposition. *Sensors* **2024**, *24*, 327. [CrossRef]
7. Xie, Y.; Liu, C.; Liu, S.; Song, W.; Fan, X. Snapshot imaging spectrometer based on pixel-level filter array (PFA). *Sensors* **2021**, *21*, 2289. [CrossRef]
8. 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.* **2016**, *33*, 95–108. [CrossRef]
9. Wagadarikar, A.; John, R.; Willett, R.; Brady, D. Single disperser design for coded aperture snapshot spectral imaging. *Appl. Optics* **2008**, *47*, B44–B51. [CrossRef]

10. Song, L.; Wang, L.; Kim, M.H.; Huang, H. High-accuracy image formation model for coded aperture snapshot spectral imaging. *IEEE Trans. Comput. Imag.* **2022**, *8*, 188–200. [CrossRef]
11. Donoho, D.L. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [CrossRef]
12. Yuan, X. Generalized alternating projection based total variation minimization for compressive sensing. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 2539–2543.
13. Lin, X.; Liu, Y.; Wu, J.; Dai, Q. Spatial-spectral encoded compressive hyperspectral imaging. *ACM Trans. Graph.* **2014**, *33*, 1–11. [CrossRef]
14. Kittle, D.; Choi, K.; Wagadarikar, A.; Brady, D.J. Multiframe image estimation for coded aperture snapshot spectral imagers. *Appl. Optics* **2010**, *49*, 6824–6833. [CrossRef] [PubMed]
15. García-Sánchez, I.; Fresnedo, Ó.; González-Coma, J.P.; Castedo, L. Coded aperture hyperspectral image reconstruction. *Sensors* **2021**, *21*, 6551. [CrossRef]
16. Xu, Y.; Wu, Z.; Chanussot, J.; Wei, Z. Hyperspectral computational imaging via collaborative Tucker3 tensor decomposition. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *31*, 98–111. [CrossRef]
17. Yang, J.; Liao, X.; Yuan, X.; Llull, P.; Brady, D.J.; Sapiro, G.; Carin, L. Compressive sensing by learning a Gaussian mixture model from measurements. *IEEE Trans. Image Process.* **2014**, *24*, 106–119. [CrossRef]
18. Liu, Y.; Yuan, X.; Suo, J.; Brady, D.J.; Dai, Q. Rank minimization for snapshot compressive imaging. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 2990–3006. [CrossRef]
19. Zhang, S.; Huang, H.; Fu, Y. Fast parallel implementation of dual-camera compressive hyperspectral imaging system. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *29*, 3404–3414. [CrossRef]
20. He, W.; Yao, Q.; Li, C.; Yokoya, N.; Zhao, Q.; Zhang, H.; Zhang, L. Non-local meets global: An iterative paradigm for hyperspectral image restoration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 2089–2107. [CrossRef]
21. Sun, Y.; Yang, Y.; Liu, Q.; Kankanhalli, M. Unsupervised spatial-spectral network learning for hyperspectral compressive snapshot reconstruction. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–14. [CrossRef]
22. Xu, P.; Liu, L.; Jia, Y.; Zheng, H.; Xu, C.; Xue, L. A Refinement Boosted and Attention Guided Deep FISTA Reconstruction Framework for Compressive Spectral Imaging. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 1–12. [CrossRef]
23. Miao, X.; Yuan, X.; Pu, Y.; Athitsos, V. l-net: Reconstruct hyperspectral images from a snapshot measurement. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 4059–4069.
24. Meng, Z.; Ma, J.; Yuan, X. End-to-end low cost compressive spectral imaging with spatial-spectral self-attention. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; Springer: Cham, Switzerland, 2020; pp. 187–204.
25. Cai, Y.; Lin, J.; Hu, X.; Wang, H.; Yuan, X.; Zhang, Y.; Timofte, R.; Van Gool, L. Mask-guided spectral-wise transformer for efficient hyperspectral image reconstruction. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 17502–17511.
26. Hu, X.; Cai, Y.; Lin, J.; Wang, H.; Yuan, X.; Zhang, Y.; Timofte, R.; Van Gool, L. Hdnet: High-resolution dual-domain learning for spectral compressive imaging. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 17542–17551.
27. 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 Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; Springer: Cham, Switzerland, 2022; pp. 686–704.
28. Cheng, Z.; Chen, B.; Lu, R.; Wang, Z.; Zhang, H.; Meng, Z.; Yuan, X. Recurrent neural networks for snapshot compressive imaging. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 2264–2281. [CrossRef] [PubMed]
29. Zheng, S.; Liu, Y.; Meng, Z.; Qiao, M.; Tong, Z.; Yang, X.; Han, S.; Yuan, X. Deep plug-and-play priors for spectral snapshot compressive imaging. *Photonics Res.* **2021**, *9*, B18–B29. [CrossRef]
30. Meng, Z.; Yu, Z.; Xu, K.; Yuan, X. Self-supervised neural networks for spectral snapshot compressive imaging. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 2622–2631.
31. Qiu, H.; Wang, Y.; Meng, D. Effective snapshot compressive-spectral imaging via deep denoising and total variation priors. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 9127–9136.
32. 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 Trans. Neural Netw. Learn. Syst.* **2023**, 1–13. [CrossRef] [PubMed]
33. 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 Trans. Image Process.* **2024**, *33*, 926–941. [CrossRef]
34. Meng, Z.; Jalali, S.; Yuan, X. Gap-net for snapshot compressive imaging. *arXiv* **2020**, arXiv:2012.08364.
35. Ma, J.; Liu, X.Y.; Shou, Z.; Yuan, X. Deep tensor admm-net for snapshot compressive imaging. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 10223–10232.

36. Wang, L.; Sun, C.; Fu, Y.; Kim, M.H.; Huang, H. Hyperspectral image reconstruction using a deep spatial-spectral prior. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 8032–8041.
37. Wang, L.; Sun, C.; Zhang, M.; Fu, Y.; Huang, H. Dnu: Deep non-local unrolling for computational spectral imaging. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 1661–1671.
38. Zhang, S.; Wang, L.; Zhang, L.; Huang, H. Learning tensor low-rank prior for hyperspectral image reconstruction. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 12006–12015.
39. Huang, T.; Dong, W.; Yuan, X.; Wu, J.; Shi, G. Deep gaussian scale mixture prior for spectral compressive imaging. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 16216–16225.
40. Huang, T.; Yuan, X.; Dong, W.; Wu, J.; Shi, G. Deep Gaussian Scale Mixture Prior for Image Reconstruction. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 10778–10794. [CrossRef]
41. Wang, L.; Wu, Z.; Zhong, Y.; Yuan, X. Snapshot spectral compressive imaging reconstruction using convolution and contextual Transformer. *Photonics Res.* **2022**, *10*, 1848–1858. [CrossRef]
42. Zhang, X.; Zhang, Y.; Xiong, R.; Sun, Q.; Zhang, J. HerosNet: Hyperspectral Explicable Reconstruction and Optimal Sampling Deep Network for Snapshot Compressive Imaging. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 17532–17541.
43. Ying, Y.; Wang, J.; Shi, Y.; Yin, B. Dual-Domain Feature Learning and Memory-Enhanced Unfolding Network for Spectral Compressive Imaging. In Proceedings of the 2023 IEEE International Conference on Multimedia and Expo (ICME), Brisbane, Australia, 10–14 July 2023; pp. 1589–1594.
44. 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. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 37749–37761.
45. Li, M.; Fu, Y.; Liu, J.; Zhang, Y. Pixel Adaptive Deep Unfolding Transformer for Hyperspectral Image Reconstruction. In Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 1–6 October 2023; pp. 12959–12968.
46. 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 Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023; pp. 22262–22271.
47. Yang, J.; Lin, T.; Liu, F.; Xiao, L. Learning Degradation-Aware Deep Prior for Hyperspectral Image Reconstruction. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 1–15. [CrossRef]
48. Xu, P.; Liu, L.; Zheng, H.; Yuan, X.; Xu, C.; Xue, L. Degradation-aware dynamic fourier-based network for spectral compressive imaging. *IEEE Trans. Multimed.* **2023**, *26*, 2838–2850. [CrossRef]
49. Qin, X.; Quan, Y.; Ji, H. Enhanced deep unrolling networks for snapshot compressive hyperspectral imaging. *Neural Netw.* **2024**, *174*, 106250. [CrossRef] [PubMed]
50. Zhang, J.; Zeng, H.; Cao, J.; Chen, Y.; Yu, D.; Zhao, Y.P. Dual Prior Unfolding for Snapshot Compressive Imaging. In Proceedings of the 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 16–22 June 2024; pp. 25742–25752.
51. 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* **2020**, arXiv:2010.11929.
52. Fu, Z.; Fu, Z.; Liu, Q.; Cai, W.; Wang, Y. Sparsett: Visual tracking with sparse transformers. *arXiv* **2022**, arXiv:2205.03776.
53. Chen, X.; Li, H.; Li, M.; Pan, J. Learning a sparse transformer network for effective image deraining. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023; pp. 5896–5905.
54. Zhou, H.; Lian, Y.; Li, J.; Liu, Z.; Cao, X.; Ma, C. Supervised-unsupervised combined transformer for spectral compressive imaging reconstruction. *Opt. Lasers Eng.* **2024**, *175*, 108030. [CrossRef]
55. Li, J.; Zheng, K.; Gao, L.; Ni, L.; Huang, M.; Chanussot, J. Model-informed Multi-stage Unsupervised Network for Hyperspectral Image Super-resolution. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5516117.
56. Cao, X.; Lian, Y.; Wang, K.; Ma, C.; Xu, X. Unsupervised hybrid network of transformer and CNN for blind hyperspectral and multispectral image fusion. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5507615. [CrossRef]
57. Ma, W.; Pan, Z.; Guo, J.; Lei, B. Achieving super-resolution remote sensing images via the wavelet transform combined with the recursive res-net. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 3512–3527. [CrossRef]
58. Dong, J.; Pan, J.; Yang, Z.; Tang, J. Multi-scale residual low-pass filter network for image deblurring. In Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 1–6 October 2023; pp. 12345–12354.
59. Torrence, C.; Compo, G.P. A practical guide to wavelet analysis. *Bull. Am. Meteorol. Soc.* **1998**, *79*, 61–78. [CrossRef]
60. Liao, X.; Li, H.; Carin, L. Generalized alternating projection for weighted-2,1 minimization with applications to model-based compressive sensing. *SIAM J. Imaging Sci.* **2014**, *7*, 797–823. [CrossRef]

61. Boyd, S.; Parikh, N.; Chu, E.; Peleato, B.; Eckstein, J. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.* **2011**, *3*, 1–122. [CrossRef]
62. 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 Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 10012–10022.
63. Beck, A.; Teboulle, M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.* **2009**, *2*, 183–202. [CrossRef]
64. Park, J.I.; Lee, M.H.; Grossberg, M.D.; Nayar, S.K. Multispectral imaging using multiplexed illumination. In Proceedings of the 2007 IEEE 11th International Conference on Computer Vision, Rio de Janeiro, Brazil, 14–21 October 2007; pp. 1–8.
65. Choi, I.; Kim, M.; Gutierrez, D.; Jeon, D.; Nam, G. High-quality hyperspectral reconstruction using a spectral prior. *ACM Trans. Graph.* **2017**, *36*, 1–13. [CrossRef]
66. Arad, B.; Timofte, R.; Yahel, R.; Morag, N.; Bernat, A.; Cai, Y.; Lin, J.; Lin, Z.; Wang, H.; Zhang, Y.; et al. Ntire 2022 spectral recovery challenge and data set. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), New Orleans, LA, USA, 19–20 June 2022; pp. 863–881.
67. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.