MST++: Multi-stage Spectral-wise Transformer for Efficient Spectral Reconstruction

Yuanhao Cai 1,*, Jing Lin 1,*, Zudi Lin 2, Haoqian Wang 1,†, Yulun Zhang 3, Hanspeter Pfister 2, Radu Timofte 3,4, Luc Van Gool 3
1 Shenzhen International Graduate School, Tsinghua University, 2 Harvard University, 3 CVL, ETH Zürich, 4 CAIDAS, JMU Würzburg

Abstract

Existing leading methods for spectral reconstruction (SR) focus on designing deeper or wider convolutional neural networks (CNNs) to learn the end-to-end mapping from the RGB image to its hyperspectral image (HSI). These CNN-based methods achieve impressive restoration performance while showing limitations in capturing the long-range dependencies and self-similarity prior. To cope with this problem, we propose a novel Transformer-based method, Multi-stage Spectral-wise Transformer (MST++), for efficient spectral reconstruction. In particular, we employ Spectral-wise Multi-head Self-attention (S-MSA) that is based on the HSI spatially sparse while spectrally self-similar nature to compose the basic unit, Spectral-wise Attention Block (SAB). Then SABs build up Single-stage Spectral-wise Transformer (SST) that exploits a U-shaped structure to extract multi-resolution contextual information. Finally, our MST++, cascaded by several SSTs, progressively improves the reconstruction quality from coarse to fine. Comprehensive experiments show that our MST++ significantly outperforms other state-of-the-art methods. In the NTIRE 2022 Spectral Reconstruction Challenge, our approach won the First place. Code and pre-trained models are publicly available at https://github.com/caiyuanhao1998/MST-plus-plus.

1. Introduction

Hyperspectral imaging records the real-world scene spectra in narrow bands, where each band captures the information at a specific spectral wavelength. Compared to normal RGB images, HSIs have more spectral bands to store richer information and delineate more details of the captured scenes. Because of this advantage, HSls have wide applications such as medical image processing [8, 51, 57], remote sensing [10, 54, 82], object tracking [35, 61], and so on. Nonetheless, such HSls with plentiful spectral information is time-consuming that spectrometers are used to scan the scenes along the spatial or spectral dimension. This limitation impedes the application scope of HSls, especially in dynamic or real-time scenes.

One way to solve this problem is to develop snapshot compressive imaging (SCI) systems and computational reconstruction algorithms [9, 12, 13, 29, 33, 47, 48, 52, 55, 56, 58, 59, 73, 74, 80] from 2D measurement to 3D HSI cube. Nevertheless, these methods rely on expensive hardware devices. To reduce costs, spectral reconstruction (SR) algorithms are proposed to reconstruct the HSI from a given RGB image, which can be easily obtained by RGB cameras.

Conventional SR methods are mainly based on sparse coding or relatively shallow learning models. Nonetheless, these model-based methods suffer from limited represent-
ing capacity and poor generalization ability. Recently, with the development of deep learning, SR has witnessed significant progress. Deep convolutional neural networks (CNNs) have been applied to learn the end-to-end mapping function from RGB images to HSI cubes. Although impressive performance have been achieved, these CNN-based methods show limitations in capturing long-range dependencies and inter-spectral self-similarity.

In recent years, the natural language processing (NLP) model, Transformer [70], has been applied in computer vision and achieved great success. The multi-head self-attention (MSA) mechanism in Transformer does better in modeling long-range dependencies and non-local self-similarity than CNN, which can alleviate the limitations of CNN-based SR algorithms. However, directly using standard Transformer [25, 49] for SR will encounter two main issues. (i) Global [25] and local [49] Transformer captures inter-actions of spatial regions. Yet, the HSI representations are spatially sparse while spectrally highly self-similar. Thus, modeling spatial inter-dependencies may be less cost-effective than capturing inter-spectra correlations. (ii) On the one hand, the computational complexity of standard global MSA is quadratic to the spatial dimension, which is a huge burden that may be unaffordable. On the other hand, local window-based MSA suffers from limited receptive fields within position-specific windows.

To address the aforementioned limitations, we propose the first Transformer-based framework, Multi-stage Spectral-wise Transformer (MST++) for efficient spectral reconstruction from RGB images. Notely, our MST++ is based on the prior work MST [13], which is customized for spectral compressive imaging restoration. Firstly, we note that HSI signals are spatially sparse while spectrally self-similar. Based on this nature, we adopt the Spectral-wise Multi-head Self-Attention (S-MSA) to compose the basic unit, Spectral-wise Attention Block (SAB). S-MSA treats each spectral feature map as a token to calculate the self-attention along the spectral dimension. Secondly, our SABs build up our proposed Single-stage Spectral-wise Transformer (SST) that exploits a U-shaped structure to extract multi-resolution spectral contextual information which is critical for HSI restoration. Finally, our MST++, cascaded by several SSTs, develops a multi-stage learning scheme to progressively improve the reconstruction quality from coarse to fine, which significantly boosts the performance.

The main contributions of this work are listed as follow.

• We propose a novel framework, MST++, for SR. To the best of our knowledge, it is the first attempt to explore the potential of Transformer in this task.

• We validate a series of natural image restoration models on this SR task. Toward them, we propose a Top-K multi-model ensemble strategy to improve the SR performance. Codes and pre-trained models of these methods are made publicly available to serve as a baseline and toolbox for further research in this topic.

• Quantitative and qualitative experiments demonstrate that our MST++ dramatically outperforms SOTA methods while requiring much cheaper Params and FLOPS. Surprisingly, our MST++ won the First place in NTIRE 2022 Spectral Reconstruction Challenge [5].

2. Related Work

2.1. Hyperspectral Image Acquisition

Traditional imaging systems for collecting HSIs often adopt spectrometers to scan the scene along the spatial or spectral dimensions. Three main types of scanners including whiskbroom scanner, pushroom scanner, and band sequential scanner are often used to capture HSIs. These scanners have been widely used in detecting, remote sensing, medical imaging, and environmental monitoring for decades. For example, pushbroom scanner and whiskbroom scanner have been used in satellite sensors [11, 64] for photogrammetric and remote sensing. However, the scanning procedure usually requires a long time, which makes it unsuitable for measuring dynamic scenes. Besides, the imaging devices are usually too large physically to be plugged in portable platforms. To address these limitations, researchers have developed SCI systems [18, 26, 50, 71, 72] to capture HSIs, where the 3D HSI cube is compressed into a single 2D measurement [81]. Among these SCI systems, coded aperture snapshot spectral imaging (CASSI) [56, 71] stands out and forms one promising research direction. Nonetheless, the SCI systems remain prohibitively expensive to date for consumer grade use. Even "low-cost" SCI systems are often in the $10K - $100K. Therefore, the SR topic has significant research and practical value.

2.2. Spectral Reconstruction from RGB

Conventional SR methods [1, 2, 34, 62, 66] are mainly based on hand-crafted hyperspectral priors. For instance, Paramar et al. [62] propose a data sparsity expending method for HSI reconstruction. Arad et al. [2] propose a sparse coding method that create a dictionary of HSI signals and their RGB projections. Aeschbacher et al. [1] suggest using relatively shallow learning models from a specific spectral prior to fulfill spectral super-resolution. However, these model-based methods suffer from limited representing capacities and poor generalization ability.

Recently, inspired by the great success of deep learning in natural image restoration [12, 24, 30, 31, 38–40, 46, 63, 92], CNNs have been exploited to learn the underlying mapping function from RGB to HSI [28, 67, 68, 78, 87]. For instance, Xiong et al. [78] propose a unified HSCNN framework for HSI reconstruction from both RGB images and compressive
measurements. Shi et al. [67] use adapted residual blocks to build up a deep residual network HSCNN-R for SR. Zhang et al. [87] customize a pixel-aware deep function-mixture network consisting to model the RGB-to-HSI mapping. However, these CNN-based SR methods achieve impressive results but show limitations in capturing non-local self-similarity and long-range inter-dependencies.

2.3. Vision Transformer

The NLP model Transformer [70] is proposed for machine translation. In recent years, it has been introduced into computer vision and gained much popularity due to its advantage in capturing long-range correlations between spatial regions. In high-level vision, Transformer has been widely applied in image classification [7, 19, 25, 27, 49, 65], object detection [6, 22, 23, 60, 86, 93], semantic segmentation [16, 69, 76, 77, 90, 91], human pose estimation [14, 15, 32, 37, 42, 43, 53, 79, 89], etc. In addition, vision Transformer has also been used in low-level vision [13, 17, 20, 24, 44, 46, 47]. For instance, Cai et al. [13] propose the first Transformer-based end-to-end framework MST for HSI reconstruction from compressive measurements. Lin et al. [47] embed the HSI sparsity into Transformer to establish a coarse-to-fine learning scheme for spectral compressive imaging. The prior work Uformer [75] adopts a U-shaped structure built up by Swin Transformer [49] blocks for natural image restoration. Nonetheless, to the best of our knowledge, the potential of Transformer in spectral super-resolution has not been explored. This work aims to fill this research gap.

3. Method

3.1. Network Architecture

As shown in Fig. 2, (a) depicts the proposed Multi-stage Spectral-wise Transformer (MST++), which is cascaded by \(N_s\) Single-stage Spectral-wise Transformers (SSTs). Our MST++ takes a RGB image as the input and reconstructs its HSI counterpart. A long identity mapping is exploited to ease the training procedure. Fig. 2 (b) shows the U-shaped SST consisting of an encoder, a bottleneck, and a decoder. The embedding and mapping block are single \(\text{conv} 3 \times 3\) layers. The embedding and mapping block are single \(\text{conv} 3 \times 3\) layers. The feature maps in the encoder sequentially undergo a downsampling operation (a strided \(\text{conv}4 \times 4\) layer), \(N_1\) Spectral-wise Attention Blocks (SABs), a downsampling operation, and \(N_2\) SABs. The bottleneck is composed of \(N_3\) SABs. The decoder employs a symmetrical architecture. The upsampling operation is a strided \(\text{deconv}2 \times 2\) layer. To avoid the information loss in the downsampling, skip connections are used between the encoder and decoder. Fig. 2 (c) illustrates the components of SAB, i.e., a Feed Forward Network (FFN as shown in Fig. 2 (d)), a Spectral-wise Multi-head Self-Attention (S-MSA), and two layer normalization. Details of S-MSA are given in Fig. 2 (e).

3.2. Spectral-wise Multi-head Self-Attention

Suppose \(\mathbf{X}_{in} \in \mathbb{R}^{H \times W \times C}\) as the input of S-MSA, which is reshaped into tokens \(\mathbf{X} \in \mathbb{R}^{H W \times C}\). Then \(\mathbf{X}\) is linearly projected into \(\text{query} \mathbf{Q} \in \mathbb{R}^{H W \times C}, \text{key} \mathbf{K} \in \mathbb{R}^{H W \times C}\), and \(\text{value} \mathbf{V} \in \mathbb{R}^{H W \times C}\):
Global MSA

We denote the input token as $X$, $W^Q$, $W^K$, and $W^V \in \mathbb{R}^{C \times C}$ are learnable parameters; biases are omitted for simplification. Subsequently, we respectively split $Q$, $K$, and $V$ into $N$ heads along the spectral channel dimension: $Q = [Q_1, \ldots, Q_N]$, $K = [K_1, \ldots, K_N]$, and $V = [V_1, \ldots, V_N]$. The dimension of each head is $d_h = \frac{C}{N}$. Please note that Fig. 2 (e) depicts the situation with $N = 1$ and some details are omitted for simplification. Different from original MSAs, our S-MSA treats each spectral representation as a token and calculates self-attention for $head_j$:

$$A_j = \text{softmax}(\sigma_j K_j^T Q_j), \quad head_j = V_j A_j,$$

where $K_j^T$ denotes the transposed matrix of $K_j$. Because the spectral density varies significantly with respect to the wavelengths, we use a learnable parameter $\sigma_j \in \mathbb{R}^1$ to adapt the self-attention $A_j$ by re-weighting the matrix multiplication $K_j^T Q_j$ inside $head_j$. Subsequently, the outputs of $N$ heads are concatenated to undergo a linear projection and then is added with a position embedding:

$$\text{S-MSA}(X) = (\text{Concat}(head_j)) W + f_p(V),$$

where $W \in \mathbb{R}^{C \times C}$ are learnable parameters, $f_p(\cdot)$ is the function to generate position embedding. It consists of two depth-wise $3 \times 3$ layers, a GELU activation, and reshape operations. The HSIs are sorted by the wavelength along the spectral dimension. Therefore, we exploit this embedding to encode the position information of different spectral channels. Finally, we reshape the result of Eq. (3) to obtain the output feature maps $X_{out} \in \mathbb{R}^{HW \times C}$.

### 3.3. Discussion with Original Transformers

In this section, we introduce the general paradigm of MSA in Transformer and then analyze the computational complexity of the spatial-wise MSAs in original Transformers and the adopted S-MSA.

#### 3.3.1 General Paradigm of MSA

We denote the input token as $X \in \mathbb{R}^{n \times C}$, where $n$ is to be determined. In spatial-wise MSAs, $n$ denotes the number of tokens. In S-MSA, $n$ represents the dimension of the token. $X$ is firstly linearly projected into query $Q \in \mathbb{R}^{n \times C}$, key $K \in \mathbb{R}^{n \times C}$, and value $V \in \mathbb{R}^{n \times C}$:

$$Q = X W^Q, \quad K = X W^K, \quad V = X W^V,$$

where $W^Q, W^K,$ and $W^V \in \mathbb{R}^{C \times C}$ are learnable parameters; biases are omitted for simplification. Subsequently, we respectively split $Q$, $K$, and $V$ into $N$ heads along the spectral channel dimension: $Q = [Q_1, \ldots, Q_N]$, $K = [K_1, \ldots, K_N]$, and $V = [V_1, \ldots, V_N]$. The dimension of each head is $d_h = \frac{C}{N}$. Please note that Fig. 2 (e) depicts the situation with $N = 1$ and some details are omitted for simplification. Different from original MSAs, our S-MSA treats each spectral representation as a token and calculates self-attention for $head_j$:

$$A_j = \text{softmax}(\sigma_j K_j^T Q_j), \quad head_j = V_j A_j,$$

where $K_j^T$ denotes the transposed matrix of $K_j$. Because the spectral density varies significantly with respect to the wavelengths, we use a learnable parameter $\sigma_j \in \mathbb{R}^1$ to adapt the self-attention $A_j$ by re-weighting the matrix multiplication $K_j^T Q_j$ inside $head_j$. Subsequently, the outputs of $N$ heads are concatenated to undergo a linear projection and then is added with a position embedding:

$$\text{S-MSA}(X) = (\text{Concat}(head_j)) W + f_p(V),$$

where $W \in \mathbb{R}^{C \times C}$ are learnable parameters. Please note that some other contents such as the position embedding are omitted for simplification. Because we only compare the main difference between original spatial-wise MSAs and S-MSA, i.e., the specific formulation of Eq. (5).

#### 3.3.2 Spatial-wise MSA

The spatial-wise MSA treats a pixel vector along the spectral dimension as a token and then calculates the self-attention for each $head_j$. Thus, Eq. (5) can be specified as

$$head_j = A_j V_j, \quad A_j = \text{softmax}(\frac{Q_j K_j^T}{\sqrt{d_h}}).$$

Eq. (7) needs to be calculated for $N$ times. Therefore, the computational complexity of spatial-wise MSA is

$$O(\text{Spatial-MSA}) = N(n^2 d_h + n^2 d_h) = 2n^2 C. \quad (8)$$

The spatial-wise MSA is mainly divided into two categories: global MSA [25] and local window-based MSA [49]. Now we analyze these two kinds of MSAs. **Global MSA.** As shown in Fig. 3 (a), global MSA samples all the tokens as key and query elements, and then calculates the self-attention. Thus, the number of tokens $n$ (key or query elements) is equal to $HW$. Then, according to Eq. (8), the computational complexity of global MSA is

$$O(\text{Global-MSA}) = 2(10W)^2 C, \quad (9)$$

which is quadratic to the spatial size of the input feature map. Global MSA enjoys a very large receptive field but its computational cost is nontrivial and sometimes unaffordable. Meanwhile, sampling redundant key elements
may easily lead to over-smooth results [41] and even non-convergence issue [93]. To cut down the computational cost, researchers propose local window-based MSA.

**Window-based MSA.** As depicted in Fig. 3 (b), W-MSA firstly splits the feature map into non-overlapping windows at size of $M^2$ and samples all the tokens inside each window to calculate self-attention. Hence, the number of tokens $n$ is equal to $M^2$ and W-MSA is conducted $HW/M^2$ times for all windows. Thus, the computational complexity is

$$O(W-MSA) = \frac{HW}{M^2} (2(M^2)^2C) = 2M^2HW C,$$

which is linear to the spatial size ($HW$). W-MSA enjoys low computational cost but suffers from limited receptive fields inside position-specific windows. As a result, some highly related non-local tokens may be neglected.

Original spatial-wise MSAs aim to capture the long-range dependencies of spatial regions. However, the HSI representations are spatially sparse while spectrally similar and correlated. Capturing spatial-wise interactions may be less cost-effective than modeling the spectral-wise correlations. Based on this HSI characteristic, we adopt S-MSA.

### 3.3.3 S-MSA

As shown in Fig. 3 (b), S-MSA treats each spectral feature map as a token and calculates the self-attention along the spectral dimension. Then Eq. (5) is specified as

$$A_j = \text{softmax}(\sigma_j K_j^T Q_j), \text{ head}_j = V_j A_j,$$

where $K_j^T$ denotes the transposed matrix of $K_j$. We note that the spectral density varies significantly with respect to the wavelengths. Therefore, we exploit a learnable parameter $\sigma_j \in \mathbb{R}^1$ to adapt the self-attention $A_j$ by re-weighting the matrix multiplication $K_j^T Q_j$ inside $\text{head}_j$. Because S-MSA treats a whole feature map as a token, the dimension of each token $n$ is equal to $HW$. Eq. (11) needs to be calculated $N$ times. Thus, the complexity of S-MSA is

$$O(S-MSA) = N (d^n_k + d^n_h) = \frac{2HW C^2}{N}.$$  

The computational complexity of W-MSA and S-MSA are linear to the spatial size ($HW$), which is much cheaper than that of global MSA (quadratic to $HW$). Nonetheless, S-MSA treats each spectral feature as a token. When calculating the self-attention $A_j$, S-MSA views the global representations and $A_j$ functions as global spatial positions. Therefore, the receptive fields of S-MSA are global and not limited to the position-specific windows.

In addition, S-MSA calculates self-attention along the spectral dimension, which is based on HSI characteristics and more suitable for HSI reconstruction when compared to spatial-wise MSAs. Thus, S-MSA is considered to be more cost-effective than global MSA and W-MSA.

For brevity, we summarize the properties of global MSA, window-based MSA, and S-MSA in Tab. 1. S-MSA enjoys global receptive fields, models the spectral-wise self-similarity, and requires linear computational costs.

### 3.4. Ensemble Strategy

In NTIRE 2022 Spectral Reconstruction Challenge, we adopt three ensemble strategies including self-ensemble, multi-scale ensemble, and Top-K multi-model ensemble to improve the performance and generality of our MST++. Now in this part, we describe them in details.

#### 3.4.1 Self-Ensemble

The RGB input is flipped up/down/left/right or rotated $90^\circ/180^\circ/270^\circ$ to be fed into the network. Subsequently, the outputs are transformed to the original state to be averaged.
3.4.2 Multi-scale Ensemble

We respectively train our models with patches at size of $256 \times 256$, $128 \times 128$, and $64 \times 64$. Then the outputs (whole images) are averaged to improve the restoration quality.

3.4.3 Top-K Multi-model Ensemble

We also train MIRNet [84], MPRNet [85], Restormer [83], HINet [21], and MST [13] families. The Top-K performers are selected for SR. Then we conduct our Top-K multi-model ensemble to fuse these reconstructed HSIs as

$$Y_{ens} = \sum_{i=1}^{K} \alpha_i \hat{Y}_i,$$

where $Y_{ens} \in \mathbb{R}^{H \times W \times N_\lambda}$ denotes the ensembled HSIs, $\hat{Y}_i$ represents the reconstructed HSIs of the $i$-th model, and $\alpha_i$ represents hyperparameter satisfying $\sum_{i=1}^{K} \alpha_i = 1$.

4. Experiment

4.1. Dataset

The dataset provided by NTIRE 2022 Spectral Reconstruction Challenge contains 1000 RGB-HSI pairs. This dataset is split into train, valid, and test subsets in proportional to 18:1:1. Each HSI at size of $482 \times 512$ has 31 wavelengths from 400 nm to 700 nm. To generate the corresponding RGB counterpart, $I \in \mathbb{R}^{H \times W \times 3}$, a transformation matrix $M \in \mathbb{R}^{N_\lambda \times 3}$ is applied to the ground-truth HSI cube $Y \in \mathbb{R}^{H \times W \times N_\lambda}$ as

$$I = Y \times M.$$  

Then the generated RGB images are injected with shot noise to simulate the real-camera situation.

4.2. Implementation Details

During the training procedure, RGB images are linearly rescaled to $[0, 1]$, after which $128 \times 128$ RGB and HSI sample pairs are cropped from the dataset. The batch size is set to 20 and the parameter optimization algorithm chooses Adam modification with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate is initialized as 0.0004 and the Cosine Annealing scheme is adopted for 300 epochs. The training data is augmented with random rotation and flipping. The proposed MST++ has been implemented on the Pytorch framework and approximately 48 hours are required for training a network on a single RTX 3090 GPU. MRAE loss function between the predicted and ground-truth HSI is adopted as the objective. In the implementation of our MST++, we set $N_s = 3$, $N_1 = N_2 = N_3 = 1$, $C = 31$.

During the testing phase, the RGB image is also linearly rescaled to $[0, 1]$ and fed into the network to fulfill the spectral recovery. Our MST++ takes 102.48 ms for per image (size $482 \times 512 \times 3$) reconstruction on a RTX 3090 GPU.

We adopt three evaluation metrics to assess the model performance. The first metric is mean relative absolute error (MRAE) that computes the pixel-wise disparity between all wavelengths of the reconstructed and ground-truth HSIs. MRAE can be formulated as

$$\text{MRAE}(Y, \hat{Y}) = \frac{1}{N} \sum_{i=1}^{N} \left\| Y[i] - \hat{Y}[i] \right\|,$$

where $\hat{Y} \in \mathbb{R}^{H \times W \times N_\lambda}$ indicates the reconstructed HSI and $N = H \times W \times N_\lambda$ denotes the number of all values on the image. The second metric is the root mean square error (RMSE) that is defined as

$$\text{RMSE}(Y, \hat{Y}) = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (Y[i] - \hat{Y}[i])^2}.$$  

Since the deciding metric for the NTIRE 2022 Spectral Reconstruction Challenge is MRAE, we directly set it as the training objective for our SR models. The last metric is the Peak Signal-to-Noise Ratio (PSNR).

4.3. Main Results

4.3.1 Quantitative Results on Valid Set

We compare our MST++ with SOTA methods including two SCI reconstruction methods (MST [13] and HDNet [29]), three SR algorithms (HSCNN+ [67], AWAN [36], and HRNet [88]), and five natural image restoration models (MIRNet [84], MPRNet [85], Restormer [83], HINet [21], EDSR [45]) on the valid set. Please note that HSCNN+ [67], AWAN [36] and HRNet [88] are the winners of NTIRE 2018 [3] and 2020 [4] Spectral Reconstruction Challenges. The results are listed in Tab. 2. Our MST++ significantly outperforms SOTA methods by a large margin while requiring the least Params and FLOPS. For instance, our MST++ achieves 3.10, 7.43, and 7.96 dB improvement in PSNR while only requiring 40.10% (1.62 / 4.04), 5.11%, 34.84% Params and 8.52% (23.05 / 270.61), 14.07%, 7.57% FLOPS when compared to AWAN, HRNet, and HSCNN+.

To intuitively show the superiority of MST++, we provide PSNR-Params-FLOPS comparisons of different algorithms in Fig. 1. The vertical axis is PSNR (performance), the horizontal axis is FLOPS (computational cost), and the circle radius is Params (memory cost). It can be seen that our MST++ takes up the top-left corner, exhibiting the extreme efficiency advantages of our method.

4.3.2 Quantitative Results on Test Set

Tab. 2 lists the top-12 leaders of NTIRE 2022 Spectral Challenge (test set), where * indicates using ensembled models. Impressively, our method won the championship out of 231 participants, suggesting the superiority of our MST++.
Table 2. Comparisons with SOTA methods on NTIRE 2022 HSI datasets (valid and test). * represents using ensembled models.

<table>
<thead>
<tr>
<th>Method</th>
<th>Params (M)</th>
<th>FLOPS (G)</th>
<th>MRAE</th>
<th>RMSE</th>
<th>PSNR</th>
<th>Username</th>
<th>MRAE</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>HSCNN+ [67]</td>
<td>4.65</td>
<td>304.45</td>
<td>0.3814</td>
<td>0.0588</td>
<td>26.36</td>
<td>pipixia</td>
<td>0.2434</td>
<td>0.0411</td>
</tr>
<tr>
<td>HRNet [88]</td>
<td>31.70</td>
<td>163.81</td>
<td>0.3476</td>
<td>0.0550</td>
<td>26.89</td>
<td>uslab</td>
<td>0.2377</td>
<td>0.0391</td>
</tr>
<tr>
<td>EDSR [45]</td>
<td>2.42</td>
<td>158.32</td>
<td>0.3277</td>
<td>0.0437</td>
<td>28.29</td>
<td>orange_dog</td>
<td>0.2377</td>
<td>0.0376</td>
</tr>
<tr>
<td>AWAN [36]</td>
<td>4.04</td>
<td>270.61</td>
<td>0.2500</td>
<td>0.0317</td>
<td>32.13</td>
<td>askldklasfj</td>
<td>0.2345</td>
<td>0.0361</td>
</tr>
<tr>
<td>HDNet [29]</td>
<td>2.66</td>
<td>173.81</td>
<td>0.2048</td>
<td>0.0274</td>
<td>32.51</td>
<td>HSHAJii</td>
<td>0.2107</td>
<td>0.0322</td>
</tr>
<tr>
<td>HiNet [21]</td>
<td>5.21</td>
<td>31.04</td>
<td>0.1890</td>
<td>0.0292</td>
<td>33.29</td>
<td>ptdoge_hot</td>
<td>0.1935</td>
<td>0.0322</td>
</tr>
<tr>
<td>MIRNet [84]</td>
<td>3.75</td>
<td>42.95</td>
<td>0.1817</td>
<td>0.0270</td>
<td>33.50</td>
<td>test_pseudo</td>
<td>0.1767</td>
<td>0.0322</td>
</tr>
<tr>
<td>Restormer [83]</td>
<td>15.11</td>
<td>93.77</td>
<td>0.1833</td>
<td>0.0274</td>
<td>33.40</td>
<td>gkdgkd</td>
<td>0.1247</td>
<td>0.0257</td>
</tr>
<tr>
<td>MPRNet [85]</td>
<td>3.62</td>
<td>101.59</td>
<td>0.1817</td>
<td>0.0270</td>
<td>33.50</td>
<td>deepplf</td>
<td>0.1767</td>
<td>0.0322</td>
</tr>
<tr>
<td>MST-L [13]</td>
<td>2.45</td>
<td>32.07</td>
<td>0.1772</td>
<td>0.0256</td>
<td>33.90</td>
<td>mialgo ls</td>
<td>0.1247</td>
<td>0.0257</td>
</tr>
<tr>
<td>MST++</td>
<td>1.62</td>
<td>23.05</td>
<td>0.1645</td>
<td>0.0248</td>
<td>34.32</td>
<td>MST++*</td>
<td>0.1131</td>
<td>0.0231</td>
</tr>
</tbody>
</table>

Figure 4. Reconstructed HSI comparisons of Scene ARAD_1k_0922 with 4 out of 31 spectral channels. 9 SOTA algorithms and our MST++ are included. The spectral curves (bottom-left) are corresponding to the selected green box of the RGB image. Please zoom in.

### 4.3.3 Qualitative Results

Fig. 4 and 5 compares the reconstructed HSIs with 4 out of 31 spectral channels of nine SOTA methods and our MST++ on the valid set. Please zoom in for a better view. The top-left part depicts the input RGB image. The right part shows the reconstructed HSI patches of the selected yellow boxes in RGB image. It can be observed that previous methods show limitations in HSI detail restoration. They either achieve over-smooth HSIs sacrificing fine-grained contents and structural details, or introduce unpleasing artifacts and blotchy textures. By contrast, MST++ does better in producing perceptually-pleasing and sharp-edge HSIs, and preserving the spatial smoothness of the homogeneous regions. This is mainly because our MST++ excels at modeling inter-spectra self-similarity and dependencies. Besides, the bottom-left part exhibits the spectral density curves corresponding to the picked region of the green box in the RGB image. The highest correlation and coincidence between our curve and the ground truth verify the spectral-wise consistency restoration effectiveness of MST++.

### 4.4. Ablation Study

we use the valid subset to conduct ablations. The baseline model is derived by removing S-MSA from MST++.

#### 4.4.1 Self-Attention Mechanism

We have discussed different self-attention mechanisms in Sec. 3.3. In this part, we conduct ablation studies to verify the performance of these MSAs including global MSA (G-MSA) [25], local window-based MSA (W-MSA) [49], Swin MSA (SW-MSA) [49], and the adopted S-MSA [13]. The results are reported in Tab. 3a. For fairness, the Params of models using different MSAs are set to the same value. Notely, the input feature of G-MSA is downscaled into $\frac{1}{4}$ size to avoid out of memory. It can be observed that our adopted S-MSA achieves the most significant improvement while requiring the least memory and computational costs. To be specific, when we respectively apply SW-MSA, W-MSA, G-MSA, and S-MSA, the performance is improved by 0.0338, 0.0553, 0.1356, and 0.1532 in MRAE while in-
### Table 3. Ablations. We train models on the train set and test on the valid set. MRAE, RMSE, Params, and FLOPS are reported.

<table>
<thead>
<tr>
<th>Method</th>
<th>Baseline</th>
<th>SW-MSA</th>
<th>W-MSA</th>
<th>G-MSA</th>
<th>S-MSA</th>
</tr>
</thead>
<tbody>
<tr>
<td>MRAE</td>
<td>0.3177</td>
<td>0.2839</td>
<td>0.2624</td>
<td>0.1821</td>
<td><strong>0.1645</strong></td>
</tr>
<tr>
<td>RMSE</td>
<td>0.0453</td>
<td>0.0399</td>
<td>0.0375</td>
<td>0.0271</td>
<td><strong>0.0248</strong></td>
</tr>
<tr>
<td>Params (M)</td>
<td>1.30</td>
<td>1.60</td>
<td>1.60</td>
<td>1.60</td>
<td>1.62</td>
</tr>
<tr>
<td>FLOPS (G)</td>
<td>17.68</td>
<td>24.10</td>
<td>24.10</td>
<td>25.11</td>
<td>23.05</td>
</tr>
</tbody>
</table>

(a) Ablation study of different self-attention mechanisms.

<table>
<thead>
<tr>
<th>$N_s$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>MRAE</td>
<td>0.1761</td>
<td>0.1716</td>
<td><strong>0.1645</strong></td>
<td>0.1711</td>
</tr>
<tr>
<td>RMSE</td>
<td>0.0266</td>
<td>0.0269</td>
<td><strong>0.0248</strong></td>
<td>0.0265</td>
</tr>
<tr>
<td>Params (M)</td>
<td>0.55</td>
<td>1.08</td>
<td>1.62</td>
<td>2.16</td>
</tr>
<tr>
<td>FLOPS (G)</td>
<td>8.10</td>
<td>15.57</td>
<td>23.05</td>
<td>30.52</td>
</tr>
</tbody>
</table>

(b) Ablation study of stage number $N_s$.

Figure 5. Reconstructed HSI comparisons of Scene ARAD_1K_0924 with 4 out of 31 spectral channels. 9 SOTA algorithms and our MST++ are included. The spectral curves (bottom-left) are corresponding to the selected green box of the RGB image. Please zoom in.

Increasing 6.42, 6.42, 7.43, and 5.37 GFLOPS. As analyzed in Sec. 3.3, these results mainly stem from the HSI spatially sparse while spectrally self-similar nature. Thus, capturing inter-spectra dependencies is more cost-effective than modeling correlations of spatial regions.

#### 4.4.2 Stage Number

We change the stage number $N_s$ of MST++ to investigate its effect. The results are shown in Tab. 3b. When $N_s = 3$, the performance achieves its peak. Therefore, we finally adopt 3-stage MST++ as our SR model.

#### 4.4.3 Ensemble Strategy

In Sec. 3.4, we adopt three ensemble strategies for NTIRE 2022 Spectral Reconstruction Challenge. In this part, we perform ablations to study their effects. On the valid set, self-ensemble, multi-scale ensemble, and Top-K (K is set to 5) multi-model ensemble respectively achieve improvements by 0.015, 0.033, and 0.045 in terms of MRAE.

#### 5. Future Work

Until now, there has not been a low-cost high-accuracy open-source baseline for SR research. Our MST++ aims to fill this gap. Moreover, all the source code and pretrained models in Tab. 2 (valid) including 11 SOTA methods are made publicly available. Our goal is to provide a model zoo and toolbox to benefit the community.

#### 6. Conclusion

In this paper, we propose the first Transformer-based framework, MST++, for spectral reconstruction from RGB. Based on the HSI spatially sparse while spectrally self-similar nature, we adopt S-MSA that treats each spectral feature map as a token for self-attention calculation to compose the basic unit SAB. Then SABs build up SST. Eventually, our MST++ is cascaded by several SSTs. Enjoying a multi-stage learning scheme, MST++ progressively improves the reconstruction quality from coarse to fine. Quantitative and qualitative experiments demonstrate that our MST++ dramatically surpasses SOTA methods while requiring cheaper memory and computational costs. Impressively, our MST++ won the First place in the NTIRE 2022 Challenge on Spectral Reconstruction from RGB.

**Acknowledgements:** This work is partially supported by the NSFC fund (61831014), the Shenzhen Science and Technology Project under Grant (ZDYBH201900000002, CJKJZD20200617102601004), the Westlake Foundation (2021B1501-2). Zudi Lin and Hanspeter Pfister acknowledge the support from NSF award IIS-2124179 and Google Cloud research credits.
References


[18] Xun Cao, Tao Yue, Xing Lin, Stephen Lin, Xin Yuan, Qionghai Dai, Lawrence Carin, and David J. Brady. Computational snapshot multispectral cameras: Toward dynamic capture of the spectral world. IEEE Signal Processing Magazine, 2016. 2


[34] Yan Jia, Yinqiang Zheng, Lin Gu, Art Subpa-Asa, Antony Lam, Yoichi Sato, and Imari Sato. From rgb to spectrum for natural scenes via manifold-based mapping. In ICCV, 2017. 2


[38] Shang Li, Guixuan Zhang, Zhengxiong Luo, and Jie Liu. Dfan: Dual feature aggregation network for lightweight image super-resolution. Wireless Communications and Mobile Computing, 2022. 2


[40] Shang Li, Guixuan Zhang, Zhengxiong Luo, Jie Liu, Zhi Zeng, and Shuwu Zhang. From general to specific: Online updating for blind super-resolution. Pattern Recognition, 2022. 2

[41] Xiangtai Li, Li Zhang, Ansheng You, Maoke Yang, Kuiyuan Yang, and Yunhai Tong. Global aggregation then local distribution in fully convolutional networks. In BMVC, 2019. 5


[48] Yang Liu, Xin Yuan, Jinli Suo, David Brady, and Qionghai Dai. Rank minimization for snapshot compressive imaging. TPAMI, 2019. 1


[50] Patrick Llull, Xuejun Liao, Xin Yuan, Jianbo Yang, David Kittle, Lawrence Carin, Guillermo Sapiro, and David J Brady. Coded aperture compressive temporal imaging. Optics Express, 2013. 2


[56] Ziyi Meng, Jiawei Ma, and Xin Yuan. End-to-end low cost compressive spectral imaging with spatial-spectral self-attention. In ECCV, 2020. 1, 2

[57] Ziyi Meng, Mu Qiao, Jiawei Ma, Zhenming Yu, Kun Xu, and Xin Yuan. Snapshot multispectral endomicroscopy. Optics Letters, 2020. 1


[59] Xin Miao, Xin Yuan, Yunchen Pu, and Vassilis Athitsos. L-net: Reconstruct hyperspectral images from a snapshot measurement. In ICCV, 2019. 1


