PanAdapter: Two-Stage Fine-Tuning with Spatial-Spectral Priors Injecting for Pansharpening

Ruocheng Wu[1*], **Zien Zhang**[1*], **Shangqi Deng**[2], **Yule Duan**[1], **Liang-Jian Deng**[1†]

[1]University of Electronic Science and Technology of China, Chengdu 611731, China
[2]Xi'an Jiaotong University, Xi'an 710049, China
wuruocheng333@outlook.com, zienzhang314@gmail.com, shangqideng0124@gmail.com,
duanyll@std.uestc.edu.cn, liangjian.deng@uestc.edu.cn

Abstract

Pansharpening is a challenging image fusion task that involves restoring images using two different modalities: low-resolution multispectral images (LRMS) and high-resolution panchromatic (PAN). Many end-to-end specialized models based on deep learning (DL) have been proposed, yet the scale and performance of these models are limited by the size of dataset. Given the superior parameter scales and feature representations of pre-trained models, they exhibit outstanding performance when transferred to downstream tasks with small datasets. Therefore, we propose an efficient fine-tuning method, namely PanAdapter, which utilizes additional advanced semantic information from pre-trained models to alleviate the issue of small-scale datasets in pansharpening tasks. Specifically, targeting the large domain discrepancy between image restoration and pansharpening tasks, the PanAdapter adopts a two-stage training strategy for progressively adapting to the downstream task. In the first stage, we fine-tune the pre-trained CNN model and extract task-specific priors at two scales by proposed Local Prior Extraction (LPE) module. In the second stage, we feed the extracted two-scale priors into two branches of cascaded adapters respectively. At each adapter, we design two parameter-efficient modules for allowing the two branches to interact and be injected into the frozen pre-trained VisionTransformer (ViT) blocks. We demonstrate that by only training the proposed LPE modules and adapters with a small number of parameters, our approach can benefit from pre-trained image restoration models and achieve state-of-the-art performance in several benchmark pansharpening datasets.

Code — https://github.com/RC-Wu/PanAdapter

Introduction

Due to the hardware limitations, existing sensors only acquire low-resolution multispectral images (LRMS) and high-resolution panchromatic (PAN) images. Pansharpening fuses LRMS and PAN images to produce high-resolution multispectral images (HRMS). Various pansharpening methods have been proposed, including traditional and deep learning-based methods. Traditional methods include component substitution (CS) techniques (Choi, Yu, and Kim 2010; Vivone 2019), multi-resolution analysis (MRA) methods (Vivone et al. 2013; Vivone, Restaino, and Chanussot 2018), and variational optimization-based (VO) approaches (Palsson, Sveinsson, and Ulfarsson 2013; He et al. 2014; Yan et al. 2022; Zhou et al. 2022). In recent years, convolutional neural networks (CNN) have been widely applied to pansharpening (Masi et al. 2016). Benefiting from the inductive bias and residual structures of CNNs, increasingly deeper networks have been proposed to address pansharpening tasks, such as PanNet (Yang et al. 2017), MSDCNN (Wei et al. 2017), BDPN (Zhang et al. 2019), DiCNN (He et al. 2019), FusionNet (Deng et al. 2020), MSDRN (Wang et al. 2021) and PMACNet (Liang et al. 2022). However, because CNNs struggle to extract global information and high-level semantic features, researchers explore methods based on Transformers. Leveraging long-range dependencies and scalability to pre-trained models and datasets, models such as full Transformers (Zhou, Liu, and Wang 2022; Meng et al. 2022) and hybrids of CNNs and Transformers (Zhu et al. 2023; Su et al. 2023; Li et al.

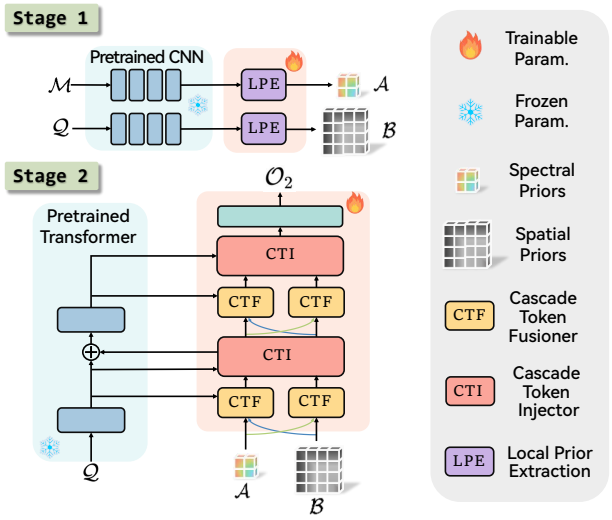


Figure 1: PanAdapter's two-stage fine-tuning framework.

*These authors contributed equally.
†Corresponding author.

2023a, 2022, 2023b) have made progress. Due to the limited size of dataset (500 to 10,000 images), blindly increasing model parameters is likely to lead to overfitting. To benefit from larger datasets and pre-trained models, we introduce the paradigm of pre-training and fine-tuning into pansharpening. By fine-tuning upstream image restoration models, we aim to enhance the performance of pansharpening tasks.

Recently, the field of image restoration has explored pre-trained models to address various image degradation problems (Chen et al. 2021; Liu et al. 2021; Wang et al. 2022b). Since pansharpening essentially is solving an inverse problem of image degradation, pre-trained image restoration models are helpful in addressing the pansharpening task. IPT (Chen et al. 2021) introduced the first Transformer-based pre-trained model for low-level tasks, followed by approaches like Swin Transformer (Liu et al. 2021), and Uformer (Wang et al. 2022b). To avoid significant parameter size and high training costs with full fine-tuning, we explore parameter-efficient fine-tuning (PEFT) strategies. In the field of PEFT, the methods of inserting small modules first emerge, such as Adapter (Houlsby et al. 2019) and LoRA (Hu et al. 2021). Prompt-based approaches (Lester, Al-Rfou, and Constant 2021; Li and Liang 2021) add a prefix prompt before the input embeddings. BitFit (Zaken, Ravfogel, and Goldberg 2021) adjusts only the bias terms of the model, while LST (Sung, Cho, and Bansal 2022) constructs side-tuning networks. In the field of computer vision, AdaptFormer (Chen et al. 2022a) adapts pre-trained ViT models, while ViT-Adapter (Chen et al. 2022b) integrates CNNs to capture local priors and compensate for ViT limitations.

However, due to two primary reasons, existing models struggle to achieve optimal performance on pansharpening tasks. Firstly, there is a significant domain gap, where models trained on natural image datasets face challenges in restoring satellite images. Secondly, conventional image restoration models lack the capability to effectively handle multi-modal inputs or multi-scale images, limiting their ability to capture the intricate spatial and spectral priors required for pansharpening. Experiments in Sec. demonstrate that existing fine-tuning strategies have shown limited performance.

Inspired by the multi-stage training strategy in image restoration domain (Zamir et al. 2021), we propose a progressive two-stage training strategy to fine-tune a pre-trained CNN model and a pre-trained ViT model addressing the above-mentioned issues. In the first stage, we fine-tune the pre-trained CNN model. Specifically, we insert the Local Prior Extraction (LPE) module to extract spatial features from PAN and spectral features from LRMS, respectively. In the second stage, we inject the priors acquired from the two branches of the CNN network output into the pre-trained ViT using a set of cascaded adapters. Furthermore, inspired by ViT-Adapter (Chen et al. 2022b), we propose two parameter-efficient interaction modules in the second stage: the Cascade Token Fusioner (CTF) module and the Cascade Token Injector (CTI) module. Referring to existing side-tuning fine-tuning methods (Sung, Cho, and Bansal 2022), our adapters adopt a cascaded architecture with intermediate results obtained from the backbone network as supplementary inputs for feature extraction, as shown in Fig. 1. The main contributions of this paper are as follows:

1. We propose PanAdapter, the first parameter-efficient fine-tuning framework designed specifically for the pansharpening task. Successfully, we apply image restoration models pre-trained on natural images to the remote sensing images. We evaluate our method on the WV3, QB and GF2 datasets, achieving state-of-the-art performance compared to various pansharpening methods.
2. We develop a novel two-stage fine-tuning strategy to reduce domain transfer difficulty and address convergence issues in the network. In the Local Prior Extraction Stage, we fine-tune a smaller pre-trained CNN network. In the Multiscale Feature Interaction Stage, we construct a dual-branch structure and fine-tune the pre-trained ViT network based on the output from the first stage.
3. We design a set of cascaded dual-branch adapters for fusing spatial and spectral priors from the first stage and injecting them into the pre-trained ViT. Through the two modules proposed in cascaded adapters, multiscale information is retained and interacted with intermediate features of the ViT backbone, which effectively balances the network's ability to extract spatial details and fuse spectral information.

Related Works

DL-Based Pansharpening

Early attempts for pansharpening primarily focus on using traditional methods (Aiazzi et al. 2002; Palsson, Sveinsson, and Ulfarsson 2013; Vivone 2019). However, these methods are often limited by their ability to extract meaningful features. PNN (Masi et al. 2016) is one of the earliest works to apply CNNs to pansharpening tasks. To boost performance, more and more researchers are working on creating larger and deeper CNN architectures for pansharpening such as PanNet (Yang et al. 2017), BDPN (Zhang et al. 2019), FusionNet (Deng et al. 2020). Subsequently, to better extract spatial domain information, researchers propose multiscale approaches, such as PMACNet (Liang et al. 2022) and BiMPan (Hou et al. 2023). Some adaptive convolution methods have also shown good performance, addressing the spatial invariance property of traditional CNNs, such as LAGConv (Jin et al. 2022), CANNet (Duan et al. 2024).

Transformers (Vaswani et al. 2017) excel in capturing global information and long-range dependencies, which are crucial for pansharpening. PanFormer (Zhou, Liu, and Wang 2022) marks the first application of Transformer in the pansharpening domain. Subsequently, researchers construct various structures to learn the global and local feature by combining Transformer and CNN, such as MHATP-Net (Zhu et al. 2023), CTCP (Su et al. 2023). However, due to the small size of the dataset, the existing models hold relatively small parameters, and are difficult to scale up to the level of visual backbones.

PEFT for Vision

Parameter-efficient fine-tuning (PEFT) aims to adapt pre-trained large models to new tasks by updating or adding

parameters. Due to the similarity between language models and computer vision, large model fine-tuning methods are quickly applied to computer vision tasks. The methods of inserting additional parameters are applied to PEFT firstly, such as Adapters (Houlsby et al. 2019). LoRA (Hu et al. 2021) adds low-rank matrices to key layers of the model to adjust its behavior. In addition, some methods select a subset of parameters from the pre-trained model for updating without altering the network structure, such as Bitfit (Zaken, Ravfogel, and Goldberg 2021). Subsequently, side network approaches emerges, which builds a ladder on top of the original large model, such as Side-Tuning (Zhang et al. 2020), LST (Sung, Cho, and Bansal 2022), where part of the output from the large model's layers serves as the input for the ladder model. Visual adapter (Chen et al. 2022a), Convpass (Jie and Deng 2022), ViT-adapter (Chen et al. 2022b) employ fine-tuning methods to effectively adapt pre-trained ViT to various image and video tasks.

Motivation

Recently, due to the lack of long-distance modeling and high-level semantic information capture capabilities of CNNs, some works have begun to attempt to use Transformers for modeling in the pansharpening task (Zhou, Liu, and Wang 2022; Meng et al. 2022). However, because of the limitation in obtaining remote sensing images, the dataset for the pansharpening task is usually small (typically ranging from 500 to 10,000 images), thus scaling up models in the pansharpening task is extremely challenging. Therefore, a natural idea is to search for pre-trained models and fine-tune on pansharpening tasks. Given the similarity in task characteristics, pre-trained image restoration models (Chen et al. 2021; Liu et al. 2021) are the preferred candidates. Nevertheless, our experiments indicate that existing fine-tuning methods perform poorly when directly adapting image restoration models to pansharpening tasks (refer to Sec.). To address existing difficulties, we propose a two-stage training framework for parameter-efficient fine-tuning. In the first stage, we fine-tune a small pre-trained CNN model by inserting the proposed Local Priors Extraction (LPE) modules to extract spatial and spectral priors at two different scales. In the second stage, based on the fine-tuned CNN model, we construct cascaded adapters for fine-tuning the pre-trained ViT model. Additionally, as the original ViT network does not contain multi-scale information, we design two branches of different scales for retaining and injecting multi-scale information tailored for pansharpening tasks. To enhance the integration efficiency of two branches, we apply the Implicit Neural Representation paradigm as decoding module at the tail end of the fusion.

Method

Overall Architecture

As illustrated in Fig. 1, the overall training process of our fine-tuning method consists of two stages.

In the first stage, we fine-tune two pre-trained CNNs trained for super-resolution (Lim et al. 2017) in parallel to obtain two different scales of feature priors. The complete process can be formulated as:

$$\mathcal{A}, \mathcal{B} = \text{SSPEN}(\mathcal{Q}, \mathcal{M}),$$

where

$$\mathcal{Q} = \text{Concat}(\mathcal{M}_{\uparrow}, \mathcal{P}).$$

$\mathcal{M} \in \mathbb{R}^{h \times w \times s}$ denotes the LRMS image, $\mathcal{M}_{\uparrow} \in \mathbb{R}^{H \times W \times s}$ denotes the upsampled LRMS image and $\mathcal{P} \in \mathbb{R}^{H \times W \times 1}$ denotes the PAN image. In addition, $\text{Concat}(\cdot, \cdot)$ means the concatenation operation in channel dimension. $\text{SSPEN}(\cdot, \cdot)$ means the Spatial-Spectral Priors Extraction Network, denoting the whole pre-trained and fine-tuning network of the first stage, which is illustrated in Fig. 2. $\mathcal{A} \in \mathbb{R}^{h \times w \times m}$ and $\mathcal{B} \in \mathbb{R}^{H \times W \times m}$ are the local spectral prior and the local spatial prior respectively, representing the results of SSPEN. Here m denotes the feature dimension of the acquired priors. After that, $\mathcal{A}$ and $\mathcal{B}$ are fed into the Tail network for decoding to obtain the predicted image of the first stage $\mathcal{O}_1$:

$$\mathcal{O}_1 = \text{Tail}_1(\mathcal{A}, \mathcal{B}),$$

where $\text{Tail}_1(\cdot, \cdot)$ denotes the Tail network of the first stage.

In the second stage, we fine-tune a ViT backbone which is pre-trained for image restoration. $\mathcal{Q}$ serves as the input of the ViT backbone. Specifically, we retain the network structure and weights of SSPEN in the second stage and inject its outputs $\mathcal{A}$ and $\mathcal{B}$ into the pre-trained ViT. The whole process can be formulated as:

$$\widehat{\mathcal{A}}, \widehat{\mathcal{B}} = \text{MFIN}(\mathcal{A}, \mathcal{B}, \mathcal{Q}),$$

where $\text{MFIN}(\cdot, \cdot, \cdot)$ means the Multiscale Feature Interaction Network, denoting the whole pre-trained and fine-tuning network of the second stage, which is illustrated in Fig. 3; $\widehat{\mathcal{A}}$ and $\widehat{\mathcal{B}}$ represent the results of MFIN at different scales. Similarly, $\widehat{\mathcal{A}}$ and $\widehat{\mathcal{B}}$ are then fed into the Tail network for decoding to obtain the predicted image of the second stage $\mathcal{O}_2$:

$$\mathcal{O}_2 = \text{Tail}_2(\widehat{\mathcal{A}}, \widehat{\mathcal{B}}),$$

where $\text{Tail}_2(\cdot, \cdot)$ denotes the Tail network of the second stage.

Local Prior Extraction Stage

Recent works (Wu et al. 2021; Park and Kim 2022; Wang et al. 2022a) have suggested that convolutions can help Transformers capture the local spatial information in a better way. Inspired by this, we propose to extract local priors from the pre-trained CNN models and inject them to the ViT blocks. The Local Prior Extraction stage network is shown in Fig. 2. To expand the applicability of our method, we aim to fine-tune a simple pre-trained network, without complicated scale changes and dense skip connections. Therefore, we choose the EDSR (Lim et al. 2017) network as the pre-trained model which is trained on the DIV2K dataset (Agustsson and Timofte 2017) for the image super-resolution task.

In order to keep the prior information at different scales, we independently fine-tune two pre-trained CNNs to obtain spectral features and spatial features from $\mathcal{M}$ and $\mathcal{Q}$, respectively. Note that the names of the features are literally for

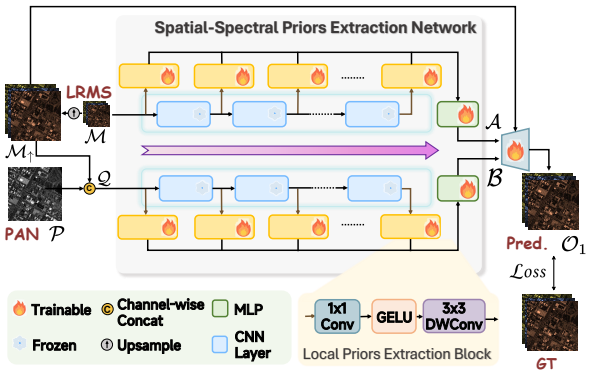


Figure 2: Network structure of the first stage, *i.e.*, the Local Prior Extraction Stage, and the details about the Local Prior Extraction (LPE) block. The frozen CNN networks are pre-trained EDSR (Lim et al. 2017).

distinguishing scales. In detail, each CNN layer uses a simple Local Prior Extraction (LPE) block to extract features with a lower channel dimension, whose structure is shown in Fig. 2. Therefore, taking the spectral branches as an example, the processing for the i-th layer's intermediate feature can be formulated as follows:

$$\widehat{\mathcal{C}}^i_{spe} = \mathrm{LPE}^i_{spe}(\mathcal{C}^i_{spe}),$$

where $\mathcal{C}^i_{spe}$ denotes the i-th layer's intermediate spectral feature, $\mathrm{LPE}^i_{spe}(\cdot)$ denotes the LPE block for processing the i-th CNN layer and $\widehat{\mathcal{C}}^i_{spe} \in \mathbb{R}^{H\times W\times d'}$ is the extracted intermediate spectral feature with a reduced channel dimension d'. In this way, we obtain $\{\widehat{C}^1_{spe}, \widehat{C}^2_{spe}, \cdots, \widehat{C}^n_{spe}\}$, which will be concatecated and down projected by a linear layer:

$$\mathcal{A} = \mathrm{Proj}_{\downarrow}(\mathrm{Concat}(\widehat{C}^1_{spe}, \widehat{C}^2_{spe}, \cdots, \widehat{C}^n_{spe})),$$

where n is the number of CNN layers and $\mathrm{Proj}_{\downarrow}(\cdot)$ denotes the down projection layer. The same process is applied to $\mathcal{C}^i_{spa}$ to obtain $\mathcal{B}$. All of the above structures are collectively referred to as the Spatial-Spectral Priors Extraction Network (SSPEN), whose structure and weights will be directly transferred to the second stage. The lightweight design of SSPEN allows us to extract priors from different network depths without fine-tuning the entire network.

In the Tail network, we apply Implicit Neural Representation (INR) paradigm (Chen, Liu, and Wang 2021; Tang, Chen, and Zeng 2021; Deng et al. 2023) to perform multi-scale feature fusion and upsampling. To capture high-frequency components in the network, we employ SIREN (Sitzmann et al. 2020) as the activation function in the INR. Following the common training strategy, the upsampled LRMS image $\mathcal{M}_{\uparrow}$ is directly added to the INR output to obtain the final prediction $\mathcal{O}_1$ of this stage, with the complete formula given as:

$$\mathcal{O}_1 = \mathrm{INR}(\mathcal{A}, \mathcal{B}) + \mathcal{M}_{\uparrow},$$

where $\mathrm{INR}(\cdot,\cdot)$ denotes the INR interpolation framework.

Multiscale Feature Interaction Stage

The network structure for the multi-scale feature interaction stage is shown in Fig. 3. In this stage, the network can be roughly divided into two parts: the ViT backbone and the cascaded adapters. To exploit the potential of plain ViT for downstream tasks, we choose the Image Processing Transformer (IPT) (Chen et al. 2021) as our ViT backbone.

The second stage network consists of cascaded identical adapters. $\mathcal{Q}$ serves as the input to the ViT backbone, while $\mathcal{A}$ and $\mathcal{B}$ serve as the inputs to the cascaded adapters. Note that due to the different scales of the inputs, the patch sizes are 4×4 for $\mathcal{Q}$ and $\mathcal{B}$ and 1×1 for $\mathcal{A}$. As in previous works (Houlsby et al. 2019; Sung, Cho, and Bansal 2022), we only fine-tune a smaller intrinsic dimension k. Therefore, $\mathcal{A}$ and $\mathcal{B}$ will be linearly projected to k dimensions as inputs to the first layer of adapter, which will be formulated as (taking $\mathcal{A}$ as an example):

$$\mathcal{F}^0_{spe} = \mathrm{Proj}_{\downarrow}(\mathrm{Reshape}(\mathcal{A})),$$

where $\mathrm{Reshape}(\cdot)$ denotes the reshape operation, $\mathcal{F}^0_{spe}$ represents the input of the first adapter.

For the j-th cascaded adapter, apart from the two outputs $\mathcal{F}^j_{spe}$ and $\mathcal{F}^j_{spe}$ of the previous adapter serving as inputs, the output of the j-th ViT blocks, *i.e.*, $\mathcal{F}^j_{vit}$, also serves as inputs. Note that a linear layer is applied in advance to obtain $\mathcal{F}^j_{vit}$, for projecting the original intermediate feature of ViT to the feature dimension k. The j-th adapter also has three outputs, $\mathcal{F}^{j+1}_{spe}$ and $\mathcal{F}^{j+1}_{spa}$, the inputs of the next layer of adapters, and $\widehat{\mathcal{F}}^j_{vit}$, which will be injected back into the ViT backbone with a linear layer projecting into original channel dimension. As is shown in Fig. 3, each adapter contains three modules, including two Cascade Token Fusioner (CTF) modules and one Cascade Token Injector (CTI) module. The two CTF modules can be separately formulated as:

$$\mathcal{F}^{j+1}_{spe} = \mathrm{CTF}^j_{spe}(\mathcal{F}^j_{spe}, \mathcal{F}^j_{spa}, \mathcal{F}^j_{vit}),$$

and

$$\mathcal{F}^{j+1}_{spa} = \mathrm{CTF}^j_{spa}(\mathcal{F}^j_{spe}, \mathcal{F}^j_{spa}, \mathcal{F}^j_{vit}),$$

where $\mathrm{CTF}^j_{spe}(\cdot,\cdot,\cdot)$ is the CTF module to obtain $\mathcal{F}^{j+1}_{spe}$ and similar to CTF^j_{spa}. The CTI module can be formulated as:

$$\widehat{\mathcal{F}}^j_{vit} = \mathrm{CTI}^j(\mathcal{F}^{j+1}_{spe}, \mathcal{F}^{j+1}_{spa}, \mathcal{F}^j_{vit}),$$

where $\mathrm{CTI}^j_{spe}(\cdot,\cdot,\cdot)$ denotes the CTI module.

The output of the last adapter layer is reshaped and fed into the Tail network for decoding. Besides, to avoid an excessively large number of parameters for fine-tuning, tokens are fed into adapters every t layers. Balancing performance and parameter count, t is ultimately set to 4. An ablation study on t is provided in the *Suppl. Mat.*

Cascade Token Fusioner Module. As shown in Fig. 3, CTF module is used to fuse the multi-scale features from two branches and interact with features from ViT. In each adapter, there are two parallel CTF modules, one for processing $\mathcal{F}^j_{spa}$ and the other for $\mathcal{F}^j_{spe}$. Taking the module for

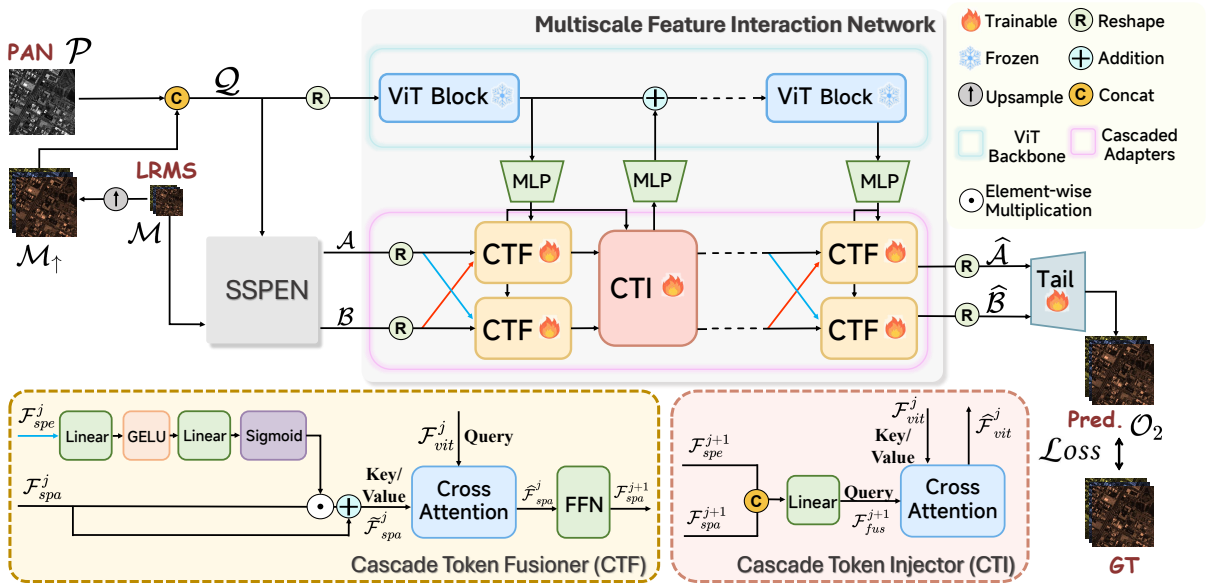


Figure 3: Network structure of the second stage, *i.e.*, the Multiscale Feature Interaction Stage, and the details about the Cascade Token Fusioner (CTF) and the Cascade Token Injector (CTI). The Spatial-Spectral Priors Extraction Network (SSPEN) and the Multiscale Feature Interaction Stage (MFIN) denote the pre-trained and fine-tuning networks of the first stage and the second stage, respectively. The frozen ViT is pre-trained Image Processing Transformer (IPT) (Chen et al. 2021).

processing $\mathcal{F}^j_{spa}$ as an example, $\mathcal{F}^j_{spe}$ goes through a proposed weighting network $w_j(\cdot)$, and the result is element-wise multiplied with $\mathcal{F}^j_{spa}$, which can be formulated as:

$$\widetilde{\mathcal{F}}^j_{spa} = \mathcal{F}^j_{spa} \cdot w_j(\mathcal{F}^j_{spe}) + \mathcal{F}^j_{spa},$$

where $\widetilde{\mathcal{F}}^j_{spa}$ serves as the Key and Value and input into a multi-head cross-attention layer. Meanwhile, $\mathcal{F}^j_{vit}$ serves as the Query of the multi-head cross-attention. The complete process can be written as:

$$\widehat{\mathcal{F}}^j_{spa} = \text{Attention}(\widetilde{\mathcal{F}}^j_{spa}, \mathcal{F}^j_{vit}) + \widetilde{\mathcal{F}}^j_{spa},$$

where Attention$(\cdot)$ denotes the multi-head cross-attention layer with 4 heads, and the specific structure of $w_j(\cdot)$ is shown in Fig. 3. Subsequently, $\widehat{\mathcal{F}}^j_{spa}$ is fed into a feed-forward network consisting of a linear layer, a GELU activation function, and another linear layer. Inspired by the Adapter structure in NLP (Houlsby et al. 2019), the intermediate dimension of two linear layers is set relatively low:

$$\mathcal{F}^{j+1}_{spa} = \widehat{\mathcal{F}}^j_{spa} + \text{FFN}(\widehat{\mathcal{F}}^j_{spa}),$$

where FFN$(\cdot)$ denotes the feed-forward network. The obtained $\mathcal{F}^{j+1}_{spa}$ will be used as the input for next adapter layer.

Cascade Token Injector Module. As shown in Fig. 3, CTI module is used to inject the spatial and spectral priors from the adapters into the ViT. Specifically, we first concatenate $\mathcal{F}^{j+1}_{spa}$ and $\mathcal{F}^{j+1}_{spe}$ along the feature dimension and project the feature dimension to k using a linear layer, yielding $\mathcal{F}^{j+1}_{fus}$ as the Query for the cross-attention:

$$\mathcal{F}^{j+1}_{fus} = \text{Linear}(\text{Concat}(\mathcal{F}^{j+1}_{spa}, \mathcal{F}^{j+1}_{spe})).$$

At the same time, the intermediate features from the i-th layer ViT, *i.e.*, $\mathcal{F}^j_{vit}$, serve as the Key and Value of the cross-attention and the whole process can be formulated as:

$$\widehat{\mathcal{F}}^j_{vit} = s_j \cdot \text{Attention}(\mathcal{F}^j_{vit}, \mathcal{F}^{j+1}_{fus}) + \mathcal{F}^j_{vit},$$

where $s_j \in \mathbb{R}$ is a trainable parameter initialized to 0. The obtained $\widehat{\mathcal{F}}^j_{vit}$ is injected into the ViT backbone.

Experiment

To validate the performance of our network, we conducte a set of experiments on different pansharpening datasets. Our results surpass the current state-of-the-art methods.

Experiment Settings

Datasets. We investigate the effectiveness of the proposed method on a wide range of datasets, including an 8-band dataset from WorldView-3 (WV3) sensor, and 4-band datasets from QuickBird (QB) and GaoFen-2 sensors. Notably, we leverage Wald's protocol to simulate the source data due to the unavailability of ground truth (GT) images.

Taking WV3 as an instance, we use 10000 PAN/LRMS/GT image pairs (64 × 64) for network training. For testing, we take 20 PAN/LRMS/GT image pairs (256 × 256) for reduced-resolution evaluation, and 20 PAN/LRMS image pairs (512 × 512) thanks to the absence of GT images on the full-resolution assessment.

Methods	Reduced-Resolution				Full-Resolution		
	PSNR	Q8	SAM	ERGAS	D_λ	D_S	QNR
PanNet	37.346 ± 2.688	0.891 ± 0.093	3.613 ± 0.766	2.664 ± 0.688	**0.0165 ± 0.0074**	0.0470 ± 0.0210	0.9374 ± 0.0271
FusionNet	38.047 ± 2.589	0.904 ± 0.090	3.324 ± 0.698	2.465 ± 0.644	0.0239 ± 0.0090	0.0364 ± 0.0137	0.9406 ± 0.0197
LAGConv	38.592 ± 2.778	0.910 ± 0.091	3.103 ± 0.558	2.292 ± 0.607	0.0368 ± 0.0148	0.0418 ± 0.0152	0.9230 ± 0.0247
PMACNet	38.595 ± 2.882	0.912 ± 0.092	3.073 ± 0.623	2.293 ± 0.532	0.0540 ± 0.0232	0.0336 ± 0.0115	0.9143 ± 0.0281
BiMPan	38.671 ± 2.732	0.915 ± 0.087	2.984 ± 0.601	2.257 ± 0.552	**0.0171 ± 0.0128**	0.0334 ± 0.0144	0.9493 ± 0.0255
HMPNet	38.684 ± 2.572	0.916 ± 0.087	3.063 ± 0.577	2.229 ± 0.545	0.0183 ± 0.0077	0.0532 ± 0.0060	0.9293 ± 0.0144
PanDiff	38.424 ± 2.686	0.898 ± 0.088	3.297 ± 0.601	2.467 ± 0.584	0.0276 ± 0.0120	0.0541 ± 0.0266	0.9201 ± 0.0364
PanMamba	38.602 ± 2.788	0.916 ± 0.090	2.940 ± 0.540	2.240 ± 0.510	0.0203 ± 0.0071	0.0422 ± 0.0141	0.9395 ± 0.0201
CANNet	**38.908 ± 2.749**	**0.920 ± 0.084**	**2.930 ± 0.593**	**2.158 ± 0.515**	0.0196 ± 0.0083	**0.0301 ± 0.0074**	**0.9510 ± 0.0132**
Ours	**39.473 ± 2.626**	**0.923 ± 0.081**	**2.917 ± 0.560**	**2.149 ± 0.492**	0.0173 ± 0.0076	**0.0304 ± 0.0086**	**0.9538 ± 0.0146**
Ideal value	$+\infty$	**1**	**0**	**0**	**0**	**0**	**1**

Table 1: The average and standard deviation calculated for all the compared approaches on 20 reduced-resolution and 20 full-resolution samples of WV3 dataset. Best results are in Red, second-best in Blue.

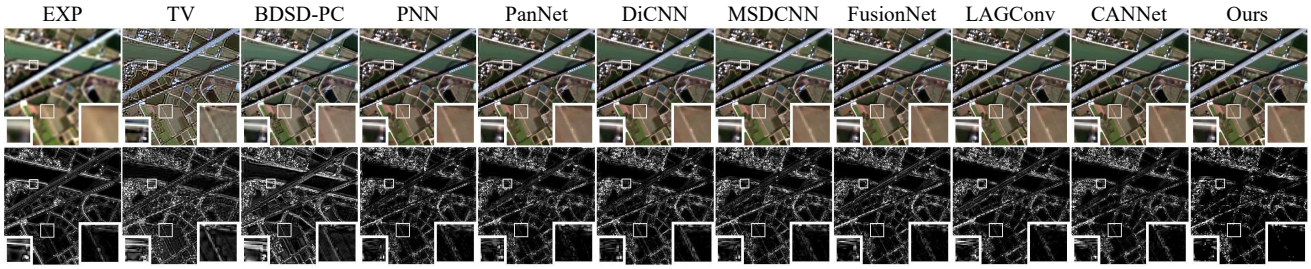


Figure 4: Qualitative evaluation result comparisons with previous pansharpening methods on GF2 reduced-resolution dataset. The first row consists of natural color output, while the second row presents the absolute error maps.

Method	SAM	ERGAS	Q4
PanNet	0.997±0.212	0.919±0.191	0.967±0.010
FusionNet	0.974±0.212	0.988±0.222	0.964±0.009
LAGConv	0.786±0.148	0.687±0.113	0.980±0.009
HMPNet	0.803±0.144	**0.562±0.107**	0.986±0.005
PanDiff	0.892±0.129	0.755±0.108	0.979±0.011
PanMamba	**0.688±0.129**	0.647±0.103	0.939±0.022
CANNet	0.707±0.148	0.630±0.128	**0.983±0.006**
Ours	**0.702±0.143**	**0.615±0.105**	**0.981±0.007**
Ideal value	**0**	**0**	**1**

Table 2: Average quantitative metrics and standard deviation on 20 reduced-resolution for the GF2 dataset. Some traditional methods and CNN methods are compared.

Method	SAM	ERGAS	Q4
PanNet	5.791±1.184	5.863±0.888	0.885±0.092
FusionNet	4.923±0.908	4.159±0.321	0.925±0.090
LAGConv	4.547±0.830	3.826±0.420	0.934±0.088
BiMPan	4.586±0.821	3.839±0.319	0.931±0.091
HMPNet	4.562±0.871	3.809±0.415	0.933±0.094
PanDiff	4.575±0.736	3.742±0.310	0.935±0.090
CANNet	**4.507±0.835**	**3.652±0.327**	**0.937±0.083**
Ours	**4.511±0.810**	**3.593±0.356**	**0.938±0.077**
Ideal value	**0**	**0**	**1**

Table 3: Average quantitative metrics and standard deviation on 20 reduced-resolution for the QB dataset. Some traditional methods and CNN methods are compared.

Benchmarks. To assess the performance of our approach, we qualitatively and quantitatively compare the proposed method with current state-of-the-art pansharpening methods, including PanNet (Yang et al. 2017), FusionNet (Deng et al. 2020), LAGConv (Jin et al. 2022), PMACNet (Liang et al. 2022), BiMPan (Hou et al. 2023), HMPNet (Tian et al. 2023), PanDiff (Meng et al. 2023), PanMamba (He et al. 2024), CANNet (Duan et al. 2024). Noted that all DL-based approaches considered for comparison are trained using the same datasets as our method, while their hyperparameter settings follow the respective original papers.

Evaluation Metrics. To assess the performance of the reduced-resolution dataset, we utilize four evaluation metrics: PSNR, Q_{2n} (Garzelli and Nencini 2009), SAM (Boardman 1993), and ERGAS (Wald 2002). For the evaluation of the full-resolution dataset, we employ three metrics: D_λ, D_s, and QNR (Vivone et al. 2014).

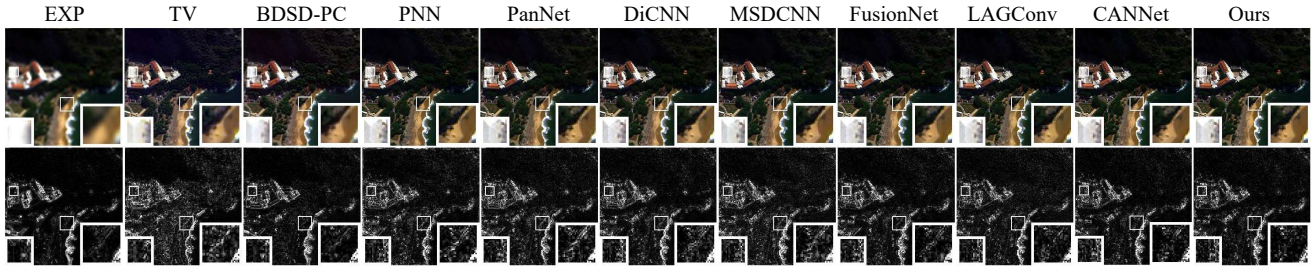


Figure 5: Qualitative evaluation result comparisons with previous pansharpening methods on WV3 reduced-resolution dataset.

Method	SAM	ERGAS	Param.(M)
VPT	-	-	0.53(0.46%)
LoRA	4.337±0.911	3.374±0.709	1.3(1.13%)
Adapter	3.809±0.844	2.793±0.654	2.07(1.79%)
LST	3.342±0.667	2.507±0.619	1.76(1.53%)
Full Fine-tune	**3.288±0.637**	**2.473±0.662**	115.2(100%)
Ours	**2.917±0.560**	**2.149±0.492**	2.19(1.85%)
Ideal value	**0**	**0**	-

Table 4: Average quantitative metrics and standard deviation of fine-tuning methods on WV3 dataset.

Method	SAM	ERGAS	Q8
w/o two-stage	3.573±0.682	2.576±0.619	0.906±0.095
Replace INR	2.957±0.640	2.381±0.916	0.916±0.090
w/o CTF module	**2.942±0.611**	**2.155±0.508**	**0.919±0.090**
w/o CTI module	3.301±0.642	2.506±0.649	0.907±0.088
Ours	**2.917±0.560**	**2.149±0.492**	**0.923±0.081**

Table 5: Ablation study on WV3 dataset.

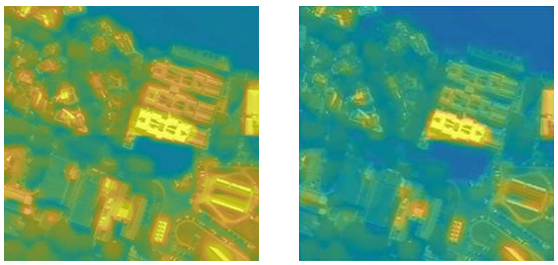
Comparison

Tabs. 1, 2 and 3 present a detailed comparison between our method and various state-of-the-art methods on three benchmark datasets: WorldView-3 (WV3), QuickBird (QB), and GaoFen-2 (GF2). The results demonstrate the robustness of PanAdapter across different datasets and its consistent capability to produce high-quality pansharpened images.

Tab. 4 compares the performance and parameter count/parameter ratio of other fine-tuning methods with our method on WV3. The specific structure of the Tail network is consistent with our method. It is worth mentioning that all existing fine-tuning methods converge slowly and are very sensitive to hyperparameters due to the lack of two-stage training and the injection of CNN priors.

Visualization

Visual comparisons depicted in Fig. 4 and 5 reveal that our method generates results that are perceptually closer to the ground truth. It can be seen from the residual images that our method recovers parts such as houses and roads very well, which is most likely due to the information of natural image datasets. Moreover, we visualize the intermediate features of the network completing the first stage of training and completing the second stage of training, respectively, as shown in Fig. 6.Lastly, the performance comparison results, as shown in Fig. 7, demonstrate that our method consistently surpasses existing state-of-the-art approaches across various levels of parametric complexity. The adjustment of the intrinsic dimension k serves as a mechanism for modifying the model's parameter count to achieve these results.



(a) (b)

Figure 6: (a) and (b) are heat maps of the intermediate features of the first and second stage networks, respectively. It can be seen that compared to CNN networks, ViT with priors injected is more accurate and efficient in capturing image semantics.

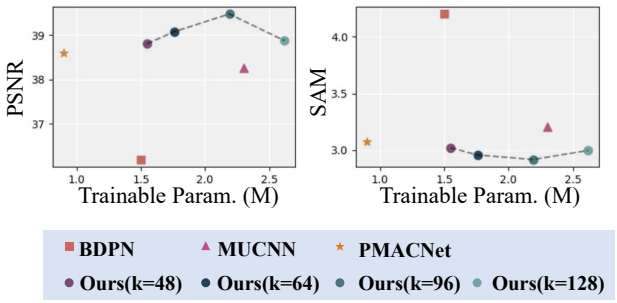


Figure 7: Comparison between SOTA methods and our method with different k, across varying parameters.

Ablation Study

The effectiveness of the two-stage fine-tuning strategy and the multiscale adapters is proven by ablation study on WV3 dataset, as shown in Tab. 5. The experiments, including stacking the network without the two-stage training strategy, replacing INR with a common convolutional upsampling module, and removing CFI and CTI modules, demonstrate

that all proposed methods positively impact performance.

Conclusion

In this paper, we propose an efficient fine-tuning method for pansharpening, *i.e.*, PanAdapter. It successfully fine-tunes pre-trained image restoration models to the pansharpening task through two-stage training and multi-scale feature extraction, thus effectively reducing the difficulty of domain transfer. The dual-branch adapters fuse and inject the spatial priors and spectral priors separately, enhancing feature extraction efficiency. Our PanAdapter outperforms state-of-the-art pansharpening methods on several datasets, providing a new paradigm for related image fusion tasks.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (Grant No. 12271083), and the Natural Science Foundation of Sichuan Province (Grant No. 2024NSFSC0038). We would like to thank Xiao Wu and Jun Ma for their valuable suggestions on the methodology of this paper, and Haoyu Deng for his insightful advice on the visualization of this work.

References

Agustsson, E.; and Timofte, R. 2017. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 126–135.

Aiazzi, B.; Alparone, L.; Baronti, S.; and Garzelli, A. 2002. Context-driven fusion of high spatial and spectral resolution images based on oversampled multiresolution analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 40(10): 2300–2312.

Boardman, J. W. 1993. Automating spectral unmixing of AVIRIS data using convex geometry concepts. In *JPL, Summaries of the 4th Annual JPL Airborne Geoscience Workshop. Volume 1: AVIRIS Workshop*.

Chen, H.; Wang, Y.; Guo, T.; Xu, C.; Deng, Y.; Liu, Z.; Ma, S.; Xu, C.; Xu, C.; and Gao, W. 2021. Pre-trained image processing transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12299–12310.

Chen, S.; Ge, C.; Tong, Z.; Wang, J.; Song, Y.; Wang, J.; and Luo, P. 2022a. Adaptformer: Adapting vision transformers for scalable visual recognition. *Advances in Neural Information Processing Systems*, 35: 16664–16678.

Chen, Y.; Liu, S.; and Wang, X. 2021. Learning continuous image representation with local implicit image function. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8628–8638.

Chen, Z.; Duan, Y.; Wang, W.; He, J.; Lu, T.; Dai, J.; and Qiao, Y. 2022b. Vision transformer adapter for dense predictions. *arXiv preprint arXiv:2205.08534*.

Choi, J.; Yu, K.; and Kim, Y. 2010. A new adaptive component-substitution-based satellite image fusion by using partial replacement. *IEEE Transactions on Geoscience and Remote Sensing*, 49(1): 295–309.

Deng, L.-J.; Vivone, G.; Jin, C.; and Chanussot, J. 2020. Detail injection-based deep convolutional neural networks for pansharpening. *IEEE Transactions on Geoscience and Remote Sensing*, 59(8): 6995–7010.

Deng, S.; Wu, R.; Deng, L.-J.; Ran, R.; and Jiang, T.-X. 2023. Implicit Neural Feature Fusion Function for Multispectral and Hyperspectral Image Fusion. *arXiv preprint arXiv:2307.07288*.

Duan, Y.; Wu, X.; Deng, H.; and Deng, L. 2024. Content-Adaptive Non-Local Convolution for Remote Sensing Pansharpening. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

Garzelli, A.; and Nencini, F. 2009. Hypercomplex quality assessment of multi/hyperspectral images. *IEEE Geoscience and Remote Sensing Letters*, 6(4): 662–665.

He, L.; Rao, Y.; Li, J.; Chanussot, J.; Plaza, A.; Zhu, J.; and Li, B. 2019. Pansharpening via detail injection based convolutional neural networks. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(4): 1188–1204.

He, X.; Cao, K.; Yan, K.; Li, R.; Xie, C.; Zhang, J.; and Zhou, M. 2024. Pan-mamba: Effective pan-sharpening with state space model. *arXiv preprint arXiv:2402.12192*.

He, X.; Condat, L.; Bioucas-Dias, J. M.; Chanussot, J.; and Xia, J. 2014. A new pansharpening method based on spatial and spectral sparsity priors. *IEEE Transactions on Image Processing*, 23(9): 4160–4174.

Hou, J.; Cao, Q.; Ran, R.; Liu, C.; Li, J.; and Deng, L.-j. 2023. Bidomain modeling paradigm for pansharpening. In *Proceedings of the 31st ACM International Conference on Multimedia*, 347–357.

Houlsby, N.; Giurgiu, A.; Jastrzebski, S.; Morrone, B.; De Laroussilhe, Q.; Gesmundo, A.; Attariyan, M.; and Gelly, S. 2019. Parameter-efficient transfer learning for NLP. In *International Conference on Machine Learning*, 2790–2799. PMLR.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Jie, S.; and Deng, Z.-H. 2022. Convolutional bypasses are better vision transformer adapters. *arXiv preprint arXiv:2207.07039*.

Jin, Z.-R.; Zhang, T.-J.; Jiang, T.-X.; Vivone, G.; and Deng, L.-J. 2022. LAGConv: Local-context adaptive convolution kernels with global harmonic bias for pansharpening. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 1113–1121.

Lester, B.; Al-Rfou, R.; and Constant, N. 2021. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*.

Li, J.; Zheng, K.; Li, Z.; Gao, L.; and Jia, X. 2023a. X-shaped interactive autoencoders with cross-modality mutual learning for unsupervised hyperspectral image super-resolution. *IEEE Transactions on Geoscience and Remote Sensing*.

Li, J.; Zheng, K.; Liu, W.; Li, Z.; Yu, H.; and Ni, L. 2023b. Model-guided coarse-to-fine fusion network for unsupervised hyperspectral image super-resolution. *IEEE Geoscience and Remote Sensing Letters*.

Li, J.; Zheng, K.; Yao, J.; Gao, L.; and Hong, D. 2022. Deep unsupervised blind hyperspectral and multispectral data fusion. *IEEE Geoscience and Remote Sensing Letters*, 19: 1–5.

Li, X. L.; and Liang, P. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

Liang, Y.; Zhang, P.; Mei, Y.; and Wang, T. 2022. PMACNet: Parallel multiscale attention constraint network for pansharpening. *IEEE Geoscience and Remote Sensing Letters*, 19: 1–5.

Lim, B.; Son, S.; Kim, H.; Nah, S.; and Mu Lee, K. 2017. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 136–144.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10012–10022.

Masi, G.; Cozzolino, D.; Verdoliva, L.; and Scarpa, G. 2016. Pansharpening by convolutional neural networks. *Remote Sensing*, 8(7): 594.

Meng, Q.; Shi, W.; Li, S.; and Zhang, L. 2023. Pandiff: A novel pansharpening method based on denoising diffusion probabilistic model. *IEEE Transactions on Geoscience and Remote Sensing*, 61: 1–17.

Meng, X.; Wang, N.; Shao, F.; and Li, S. 2022. Vision transformer for pansharpening. *IEEE Transactions on Geoscience and Remote Sensing*, 60: 1–11.

Palsson, F.; Sveinsson, J. R.; and Ulfarsson, M. O. 2013. A new pansharpening algorithm based on total variation. *IEEE Geoscience and Remote Sensing Letters*, 11(1): 318–322.

Park, N.; and Kim, S. 2022. How do vision transformers work? *arXiv preprint arXiv:2202.06709*.

Sitzmann, V.; Martel, J.; Bergman, A.; Lindell, D.; and Wetzstein, G. 2020. Implicit neural representations with periodic activation functions. *Advances in Neural Information Processing Systems*, 33: 7462–7473.

Su, Z.; Yang, Y.; Huang, S.; Wan, W.; Tu, W.; Lu, H.; and Chen, C. 2023. CTCP: Cross Transformer and CNN for Pansharpening. In *Proceedings of the 31st ACM International Conference on Multimedia*, 3003–3011.

Sung, Y.-L.; Cho, J.; and Bansal, M. 2022. Lst: Ladder side-tuning for parameter and memory efficient transfer learning. *Advances in Neural Information Processing Systems*, 35: 12991–13005.

Tang, J.; Chen, X.; and Zeng, G. 2021. Joint implicit image function for guided depth super-resolution. In *Proceedings of the 29th ACM International Conference on Multimedia*, 4390–4399.

Tian, X.; Li, K.; Zhang, W.; Wang, Z.; and Ma, J. 2023. Interpretable model-driven deep network for hyperspectral, multispectral, and panchromatic image fusion. *IEEE Transactions on Neural Networks and Learning Systems*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Vivone, G. 2019. Robust band-dependent spatial-detail approaches for panchromatic sharpening. *IEEE transactions on Geoscience and Remote Sensing*, 57(9): 6421–6433.

Vivone, G.; Alparone, L.; Chanussot, J.; Dalla Mura, M.; Garzelli, A.; Licciardi, G. A.; Restaino, R.; and Wald, L. 2014. A critical comparison among pansharpening algorithms. *IEEE Transactions on Geoscience and Remote Sensing*, 53(5): 2565–2586.

Vivone, G.; Restaino, R.; and Chanussot, J. 2018. Full scale regression-based injection coefficients for panchromatic sharpening. *IEEE Transactions on Image Processing*, 27(7): 3418–3431.

Vivone, G.; Restaino, R.; Dalla Mura, M.; Licciardi, G.; and Chanussot, J. 2013. Contrast and error-based fusion schemes for multispectral image pansharpening. *IEEE Geoscience and Remote Sensing Letters*, 11(5): 930–934.

Wald, L. 2002. *Data fusion: definitions and architectures: fusion of images of different spatial resolutions*. Presses des MINES.

Wang, W.; Xie, E.; Li, X.; Fan, D.-P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; and Shao, L. 2022a. Pvt v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, 8(3): 415–424.

Wang, W.; Zhou, Z.; Liu, H.; and Xie, G. 2021. MSDRN: Pansharpening of multispectral images via multi-scale deep residual network. *Remote Sensing*, 13(6): 1200.

Wang, Z.; Cun, X.; Bao, J.; Zhou, W.; Liu, J.; and Li, H. 2022b. Uformer: A general u-shaped transformer for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17683–17693.

Wei, Y.; Yuan, Q.; Meng, X.; Shen, H.; Zhang, L.; and Ng, M. 2017. Multi-scale-and-depth convolutional neural network for remote sensed imagery pan-sharpening. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 3413–3416. IEEE.

Wu, H.; Xiao, B.; Codella, N.; Liu, M.; Dai, X.; Yuan, L.; and Zhang, L. 2021. Cvt: Introducing convolutions to vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 22–31.

Yan, K.; Zhou, M.; Huang, J.; Zhao, F.; Xie, C.; Li, C.; and Hong, D. 2022. Panchromatic and multispectral image fusion via alternating reverse filtering network. *Advances in Neural Information Processing Systems*, 35: 21988–22002.

Yang, J.; Fu, X.; Hu, Y.; Huang, Y.; Ding, X.; and Paisley, J. 2017. PanNet: A deep network architecture for pansharpening. In *Proceedings of the IEEE International Conference on Computer Vision*, 5449–5457.

Zaken, E. B.; Ravfogel, S.; and Goldberg, Y. 2021. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *arXiv preprint arXiv:2106.10199*.

Zamir, S. W.; Arora, A.; Khan, S.; Hayat, M.; Khan, F. S.; Yang, M.-H.; and Shao, L. 2021. Multi-stage progressive image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14821–14831.

Zhang, J. O.; Sax, A.; Zamir, A.; Guibas, L.; and Malik, J. 2020. Side-tuning: a baseline for network adaptation via additive side networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 698–714. Springer.

Zhang, Y.; Liu, C.; Sun, M.; and Ou, Y. 2019. Pan-sharpening using an efficient bidirectional pyramid network. *IEEE Transactions on Geoscience and Remote Sensing*, 57(8): 5549–5563.

Zhou, H.; Liu, Q.; and Wang, Y. 2022. Panformer: A transformer based model for pan-sharpening. In *2022 IEEE International Conference on Multimedia and Expo (ICME)*, 1–6. IEEE.

Zhou, M.; Huang, J.; Yan, K.; Yang, G.; Liu, A.; Li, C.; and Zhao, F. 2022. Normalization-based feature selection and restitution for pan-sharpening. In *Proceedings of the 30th ACM International Conference on Multimedia*, 3365–3374.

Zhu, W.; Li, J.; An, Z.; and Hua, Z. 2023. Mutiscale hybrid attention transformer for remote sensing image pansharpening. *IEEE Transactions on Geoscience and Remote Sensing*, 61: 1–16.