
SSDiff: Spatial-spectral Integrated Diffusion Model for Remote Sensing Pansharpening

Yu Zhong[†]

University of Electronic Science
and Technology of China
yuuzhong1011@gmail.com

Xiao Wu[†]

University of Electronic Science
and Technology of China
wxwsx1997@gmail.com

Zihan Cao

University of Electronic Science
and Technology of China
iamzihan666@gmail.com

Hong-Xia Dou

Xihua University
hongxiadou1991@126.com

Liang-Jian Deng^{*}

University of Electronic Science and Technology of China
liangjian.deng@uestc.edu.cn

Abstract

Pansharpening is a significant image fusion technique that merges the spatial content and spectral characteristics of remote sensing images to generate high-resolution multispectral images. Recently, denoising diffusion probabilistic models have been gradually applied to visual tasks, enhancing controllable image generation through low-rank adaptation (LoRA). In this paper, we introduce a spatial-spectral integrated diffusion model for the remote sensing pansharpening task, called SSDiff, which considers the pansharpening process as the fusion process of spatial and spectral components from the perspective of subspace decomposition. Specifically, SSDiff utilizes spatial and spectral branches to learn spatial details and spectral features separately, then employs a designed alternating projection fusion module (APFM) to accomplish the fusion. Furthermore, we propose a frequency modulation inter-branch module (FMIM) to modulate the frequency distribution between branches. The two components of SSDiff can perform favorably against the APFM when utilizing a LoRA-like branch-wise alternative fine-tuning method. It refines SSDiff to capture component-discriminating features more sufficiently. Finally, extensive experiments on four commonly used datasets, i.e., WorldView-3, WorldView-2, GaoFen-2, and QuickBird, demonstrate the superiority of SSDiff both visually and quantitatively. The code is available at https://github.com/Z-ynos/SSdiff_main.

1 Introduction

Due to physical limitations, satellite sensors cannot directly acquire high-resolution multispectral images (HrMSI). Instead, they can obtain high-resolution panchromatic (PAN) images and low-resolution multispectral images (LrMSI). Pansharpening techniques can merge PAN images with LrMSI, generating HrMSI that possess both high spatial and spectral resolutions. Pansharpening,

[†]Equal contribution.

^{*}Corresponding author.

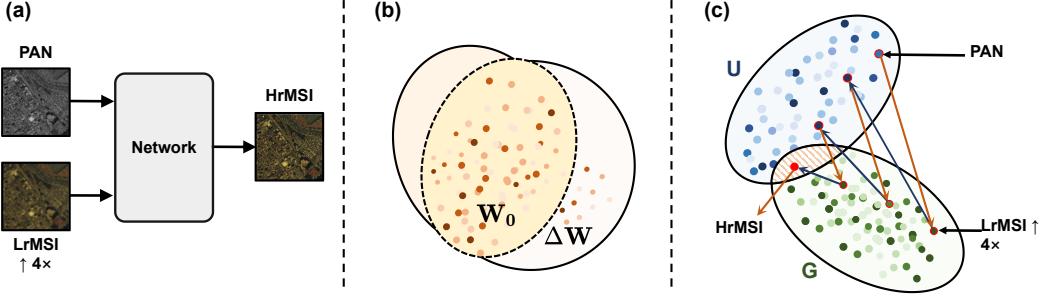


Figure 1: Schematic of (a) DL-based pansharpening approach in a supervised fashion, in which the “network” can be any deep module, e.g., denoising diffusion probabilistic models (DDPM). The comparison of (b) the LoRA based on DDPM and (c) the proposed APFM in our SSDiff. G and U represent the spectral and spatial domains, respectively. The LoRA can expand learnable weights W_0 with ΔW (but without applications to pansharpening), and the given APFM can obtain pansharpened HrMSI from PAN image and LrMSI through alternating projections.

as a fundamental preprocessing method, has been widely utilized in various applications, including change detection [37] and image segmentation [45].

Pansharpening methods are roughly categorized into four types: component substitution (CS) methods, multi-resolution analysis (MRA) methods, variational optimization (VO) techniques, and deep learning (DL) methods, as shown in Fig. 1 (a). The CS method [17, 22] involves projecting the LrMSI into a specific domain and replacing the spatial components of the LrMSI in that domain with the corresponding components from the PAN image. CS-based methods can generate fusion images with high spatial fidelity but spectral distortion with fast runtime. MRA-based methods [24, 32] extract spatial details from the PAN image through multiscale decomposition and inject them into the LrMSI. While MRA-based methods effectively preserve spectral information, they may sacrifice spatial details. Compared to CS-based and MRA-based methods, VO-based techniques [40, 41] gain more mathematical guarantees but require handling a higher computational burden and more adjustable parameters.

In recent years, DL-based methods [12, 39, 42, 38, 8, 19] have increasingly been applied across various fields, including pansharpening tasks, yielding exciting results. Traditional DL-based methods typically utilize a single-scale model to process information from PAN images and LrMSI. In a single-scale network, PAN images and LrMSI are usually stacked together without distinguishing the information contained within them, serving as inputs to the network. They overlook the disparities in the deep-level information inherent in both, potentially leading to the omission of crucial discriminative features and subsequently influencing lower fusion performance. Then, dual-branch methods [6, 18] based on deep learning can differentiate and hierarchically learn information from PAN and LrMSI. Thanks to this design, it has shown outstanding performance in pansharpening tasks. However, the cumbersome structure of dual-branch networks makes it challenging to perform localized fine-tuning. Denoising diffusion probabilistic model (DDPM) [13] is attaining attention in remote sensing pansharpening [21, 4]. Unfortunately, existing DDPM-based methods have not yet designed models for the discriminative features required in the pansharpening task.

Considering the characteristics of the pansharpening task, we propose a novel SSDiff method based on subspace decomposition, which leverages spatial and spectral branches to discriminatively capture global spatial information and spectral features, respectively. Additionally, we further construct an alternating projection fusion module (APFM) to fuse the captured spatial and spectral components. Besides, a frequency modulation inter-branch module (FMIM) is designed to overcome the problem of uneven distribution of frequency information between two branches in the denoising process. Finally, through the proposed LoRA-like branch-wise alternating fine-tuning (L-BAF), our SSDiff can further reveal spatial and spectral information not discovered in each branch. The main **contributions** of this work are as follows:

- Our SSDiff is based on subspace decomposition to divide the network into spatial and spectral branches. In addition, for subspace decomposition, we give an illustration of vector projection and construct an alternating projection fusion module (APFM). APFM transforms the process of fusing HrMSI into the fusion process of spatial and spectral components. Moreover, our SSDiff is tested on four widely used pansharpening datasets and achieves state-of-the-art (SOTA) performance.
- The frequency modulation inter-branch module is used at the junction of spectral and spatial branches to enrich extracted spatial information with more high-frequency information in the denoising process.
- The proposed L-BAF method is used to fine-tune the network based on the proposed APFM, where the spatial and spectral branches are updated alternately. This design allows us to alternately fine-tune the two branches without increasing the parameter count, enabling the learning of more discriminative features.

2 Related Works

DL-based Methods. As a simple but effective method, the representative single-scale coupling model, namely PNN [20], first proposes a simple and effective three-layer CNN architecture and achieves the best results at that time. Subsequently, other methods such as FusionNet [5], DCFNet [39], and others adopt similar coupled input approaches and successfully design their networks. However, these methods still have significant room for improvement in spectral fidelity and generalization performance due to weak feature representation in their network structure designs. In multi-source image fusion tasks, images acquired from diverse sources exhibit varying characteristics. Coupling two information sources together may suffer from inadequate feature extraction. Then, spatial and spectral branches methods [6, 18, 25] based on deep learning can differentiate and hierarchically learn information from PAN images and LrMSI. These methods can better exploit the potential advantages of multi-scale information.

Diffusion-based Model. DDPM, as a generative model, has been widely applied in various domains such as text-to-image generation [26], image editing [16] and image classification [43]. In recent years, DDPM has shown its prominence in image processing tasks [9, 28, 44]. Among them, Song et al. [28] propose denoising diffusion implicit models (DDIM), where they design a non-Markov chain sampling process, accelerating the sampling of diffusion models. Then, through some simple modifications, IDDPM [23] achieves competitive log-likelihoods while preserving the high sample quality of DDPM. Currently, DDPM is attracting attention in the field of pansharpening [21, 4]. These DDPM-based methods treat PAN and LrMSI as model fusion conditions, unlike other pansharpening methods where they serve as fusion targets.

LoRA: Low-rank Adaptation of Large Language Models. For the fine-tuning of parameters in large pre-trained models, Hu et al. [14] introduce the LoRA to freeze the pre-trained model weights and inject trainable low-rank decomposition matrices into each layer of the Transformer architecture. This significantly reduces the number of trainable parameters for downstream tasks. For $\mathbf{H} = \mathbf{W}_0 \mathbf{X}$, the modified forward pass of the LoRA follows the formula:

$$\mathbf{H} = \mathbf{W}_0 \mathbf{X} + \Delta \mathbf{W} \mathbf{X} = \mathbf{W}_0 \mathbf{X} + \mathbf{B} \mathbf{A} \mathbf{X}, \quad (1)$$

where $\mathbf{W}_0 \in \mathbb{R}^{d \times k}$ is a pre-trained weight matrix, $\mathbf{X} \in \mathbb{R}^{k \times n}$, $\mathbf{B} \in \mathbb{R}^{d \times r}$, $\mathbf{A} \in \mathbb{R}^{r \times k}$, and the rank $r \ll \min(d, k)$. Actually, $\mathbf{W}_0 + \Delta \mathbf{W} = \mathbf{W}_0 + \mathbf{B} \mathbf{A}$ represents a low-rank decomposition.

Motivation. PAN images and LrMSI are obtained from different sensors and contain distinct feature information. PAN images exhibit richer spatial details, while LrMSI possesses more abundant spectral information. However, existing DDPM-based methods have not yet designed models specifically for the discriminative features required in the pansharpening task. As a result, these methods suffer from

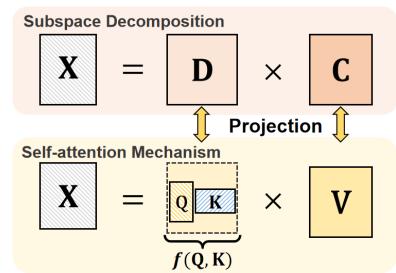


Figure 2: Schematic diagram of the relationship between subspace decomposition and self-attention mechanism. $f(\mathbf{Q}, \mathbf{K})$ is the classic self-similarity equation in self-attention mechanism.

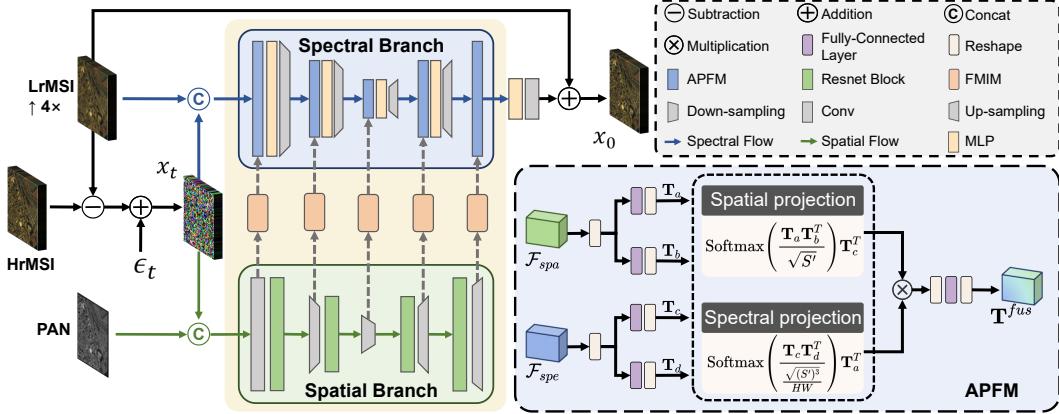


Figure 3: Overall framework of the proposed SSDiff. $\epsilon_t = \sqrt{1 - \bar{\alpha}_t} \epsilon$ is a Gaussian noise, where t is the time step. \mathcal{F}_{spa} is the output of the spatial branch, and \mathcal{F}_{spe} is the output of the spectral branch. The process of APFM follows Theorem 1.

issues such as insufficient feature learning and generalization capabilities, though they may employ low-rank adaptation (LoRA) to improve the performance of DDPM for the pansharpening task.

To alleviate these problems, we propose SSDiff, which transforms the problem of solving HrMSI into a fusion problem of spatial and spectral components. Significantly, we give an illustration of linear algebra to remove the gap between subspace decomposition and the self-attention mechanism. The SSDiff utilizes vector projection to discriminatively capture global spatial information and spectral features in spatial and spectral branches. By introducing subspace decomposition, we can further illustrate and generalize the vector projection to the matrix form. Based on this, we propose an APFM that naturally decouples spatial and spectral information and fuses the captured features. Unlike the LoRA method, the APFM can establish low-rank representations for the spatial-spectral branches more accurately, as shown in Fig. 1. Furthermore, the spectral branch contains abundant low-frequency information. When low- and high-frequency information from the spatial branch is injected into the spectral branch, it may result in an overemphasis on low-frequency information, impacting the denoising performance of the model. Based on this, we propose FMIM to modulate the frequency information between different branches. The overall model is trained by L-BAF to uncover spatial and spectral information not discovered in each branch.

3 Methodology

3.1 SSDiff Architecture

Inspired by the LoRA approach, we view the pansharpening task as the fusion of spatial and spectral components, where the spatial and spectral elements can be considered as a matrix decomposition of a multi-spectral image. Based on these characteristics, our SSDiff employs a model comprising a spatial branch and a spectral branch, as shown in Fig. 3. Both the spatial branch and the spectral branch comprise two encoder layers and two decoder layers. Down-sampling occurs between the encoder layers to decrease the spatial resolution while increasing the number of channels. Up-sampling operation between the two layers of the decoder increases the spatial resolution while decreasing the number of channels, and a down-sampling convolution layer connects the encoder and the decoder. The spatial branch employs ResNet [11] blocks to convert spatial images into features. These spatial features are transmitted to corresponding layers in the spectral branch via a frequency modulation inter-branch module. Additionally, fusing incoming spatial features and spectral information via an alternating projection fusion module.

Eventually, it is delivered to the next stage via an MLP. In this work, we convert the objective from ϵ to x_0 , so the loss function \mathcal{L}_{simple} [4] takes the following form:

$$\mathcal{L}_{simple} = \mathbb{E} [\|x_0 - x_\theta(\mathbf{x}_t, \mathbf{c}, t)\|_1], \quad (2)$$

where x_θ denotes the prediction of the model and \mathbf{c} is the conditions for injecting the model. Inspired by FusionNet [5], our SSDiff changes the forward and backward denoising objects during the training process from HrMSI to the difference between HrMSI and up-sampled LrMSI. The detailed training process of SSDiff can be found in Appendix. D.

3.2 Alternating Projection Fusion

This section starts with vector projection in linear algebra (See Lemma 1) and subspace decomposition (See Definition 1) to illustrate vector projection as a specific subspace decomposition. Then, the self-attention mechanism [30] is generalized into the proposed alternating projection fusion framework, i.e., APFM. In what follows, we rewrite vector projection as follows.

Lemma 1 ([29]). *Assuming that the existing two arbitrary vectors $\mathbf{a} \in \text{dom}\mathbf{U} \in \mathbb{R}^n$ and $\mathbf{b} \in \text{dom}\mathbf{G} \in \mathbb{R}^n$, then $\mathbf{Pb} = \lambda\mathbf{a} = \mathbf{p}$, we have the following formula:*

$$\mathbf{p} = \frac{\mathbf{aa}^T}{\mathbf{a}^T \mathbf{a}} \mathbf{b}. \quad (3)$$

where \mathbf{P} is projection matrix, λ is the scaling factor, and \mathbf{p} is the vector in the same domain as \mathbf{a} .

Proof. For any two vectors \mathbf{a} and \mathbf{b} , there exists a vector $\mathbf{e} = \mathbf{p} - \mathbf{b}$ such that \mathbf{e} is orthogonal to \mathbf{a} . We have the following equation:

$$\mathbf{a}^T \mathbf{e} = \mathbf{a}^T (\mathbf{p} - \mathbf{b}) = \mathbf{a}^T (\lambda\mathbf{a} - \mathbf{b}) = 0, \quad (4)$$

thus, we have

$$\lambda = \frac{\mathbf{a}^T \mathbf{b}}{\mathbf{a}^T \mathbf{a}}. \quad (5)$$

Taking Eq. (5) into $\mathbf{Pb} = \lambda\mathbf{a} = \mathbf{p}$, we have the conclusion:

$$\mathbf{p} = \mathbf{Pb} = \mathbf{a}\lambda = \frac{\mathbf{aa}^T}{\mathbf{a}^T \mathbf{a}} \mathbf{b}. \quad (6)$$

□

Definition 1 ([7]). *Assume that $\mathbf{D} \in \mathbb{R}^{S \times L}$ is the subspace and $\mathbf{C} \in \mathbb{R}^{L \times HW}$ is the corresponding coefficients. We have:*

$$\mathbf{Z} = \mathbf{DC}. \quad (7)$$

Based on subspace decomposition, we can take spatial and spectral components to accomplish the pansharpening of remote sensing images. According to Lemma 1, we can further determine a specific subspace decomposition in Definition 1. The subspace represents the projection relationship between vectors \mathbf{a} and \mathbf{b} . Interestingly, we find that we can generalize Eq. (3) and Eq. (7) to the matrix form of the self-attention mechanism, as shown in Fig. 2. In other words, the self-attention mechanism is represented as the vector projection of Eq. (3), which is the low-rank subspace decomposition.

Remark 1. *Back to the pansharpening applications, the input PAN and LrMSI are mapped to a high-dimensional feature space. The features often exhibit significant correlations between frequency bands, while spectral vectors typically reside in a low-dimension subspace. These features can be represented as Eq. (7). In this way, the characteristics in the image domain can be transformed into the subspace.*

Based on the above analysis, we can build an alternating projection framework, which is summarized in the following theorem.

Theorem 1. *Assuming that $\mathcal{F}_{spa} \in \mathbb{R}^{H \times W \times S}$ and $\mathcal{F}_{spe} \in \mathbb{R}^{H \times W \times S}$ from the spatial and spectral branches, they can be alternatively projected as follows:*

$$\mathbf{T}^{spa} = \text{Softmax} \left(\frac{\mathbf{T}_a \mathbf{T}_b^T}{\sqrt{S'}} \right) \mathbf{T}_c^T, \quad \mathbf{T}^{spe} = \text{Softmax} \left(\frac{\mathbf{T}_c \mathbf{T}_d^T}{\frac{\sqrt{(S')^3}}{HW}} \right) \mathbf{T}_a^T, \quad (8)$$

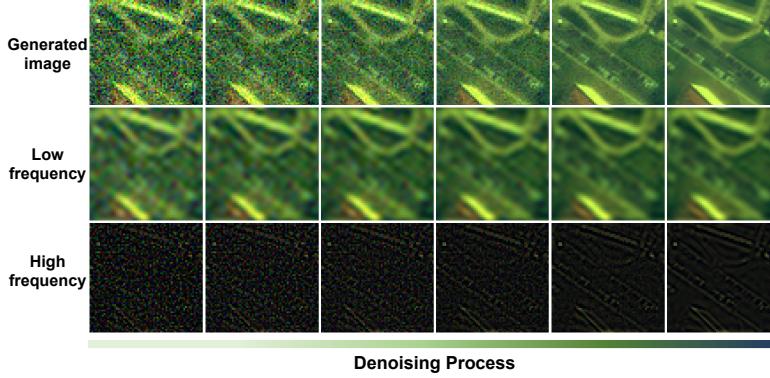


Figure 4: The denoising process. The top row consists of a series of iteratively generated images from the gradual denoising process. The subsequent two rows represent the associated low-frequency and high-frequency spatial domain information obtained through inverse Fourier transform from the denoised image in the first row of each corresponding step.

where $\mathbf{T}^{spa} \in \mathbb{R}^{HW \times S'}$ and $\mathbf{T}^{spe} \in \mathbb{R}^{S' \times HW}$ denote the features of spatial domain and spectral domain separately. $\mathbf{T}_a \in \mathbb{R}^{HW \times S'}$ and $\mathbf{T}_b \in \mathbb{R}^{HW \times S'}$ are features in the spatial domain generated by \mathcal{F}_{spa} , where S' is the channel of self-attention. $\mathbf{T}_c \in \mathbb{R}^{S' \times HW}$ and $\mathbf{T}_d \in \mathbb{R}^{S' \times HW}$ are features in the spectral domain generated by \mathcal{F}_{spe} . $\sqrt{S'}$ and $\frac{\sqrt{(S')^3}}{HW}$ are constants related to the matrix size. $\text{Softmax}(\cdot)$ stands for the Softmax function.

Proof. According to Lemma 1, we can generalize the self-attention mechanism [30], where \mathbf{Q} and \mathbf{K} are the features from domU , and \mathbf{V} is the feature from domG , respectively. Thus, we have:

$$\text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right) \Rightarrow \frac{\mathbf{a}\mathbf{a}^T}{\mathbf{a}^T\mathbf{a}}, \quad \mathbf{V} \Rightarrow \mathbf{b}, \quad (9)$$

where " \Rightarrow " suggests that the left and right parts of the equation are similar in form, and the left side is a special case of the right side. Through this equation, we can partially explain the principle of cross-attention using the vector projection theorem. Then we transform the projection relationship between the spatial domain (domU) and spectral domain (domG), where $\mathbf{T}_a, \mathbf{T}_b \in \text{domU}$ and $\mathbf{T}_c, \mathbf{T}_d \in \text{domG}$. d is a self-attention constant. As a result, the alternating projection is complete from the spatial/spectral domain to the spectral/spatial domain, i.e., Eq. (8). \square

In addition, we need to get fused outputs from \mathbf{T}^{spa} and \mathbf{T}^{spe} . Without loss of generality, we have

$$\mathbf{T}^{fus} = \mathbf{T}^{spa} \odot \mathbf{T}^{spe}, \quad (10)$$

where \odot defines element-wise multiplication. Element-wise multiplication is used to fuse spatial and spectral information to obtain $\mathbf{T}^{fus} \in \mathbb{R}^{HW \times S'}$.

Comparing Eq. (3) with Eq. (8), this subspace is built by vector projection and naturally decouples spatial and spectral information into the self-attention mechanism. This inspires us to apply a fine-tuning method similar to LoRA methods (See details in Sect. 3.4).

3.3 Frequency Modulation Inter-branch Module

Through the APFM, we build an effective fusion module from the characteristics of images. Interestingly, there are some differences between the spatial and spectral components. The spectral branch contains abundant low-frequency information. When low- and high-frequency information from the spatial branch is injected into the spectral branch, it may result in an overemphasis on low-frequency information, impacting the denoising performance of the model. We found that modulating the frequency distribution contributes to SSDiff obtaining better fusion results.

As shown in Fig. 4, the low-frequency components undergo a gradual modulation characterized by a slow and subtle rate of change in the denoising process. In contrast, the modulation process of the

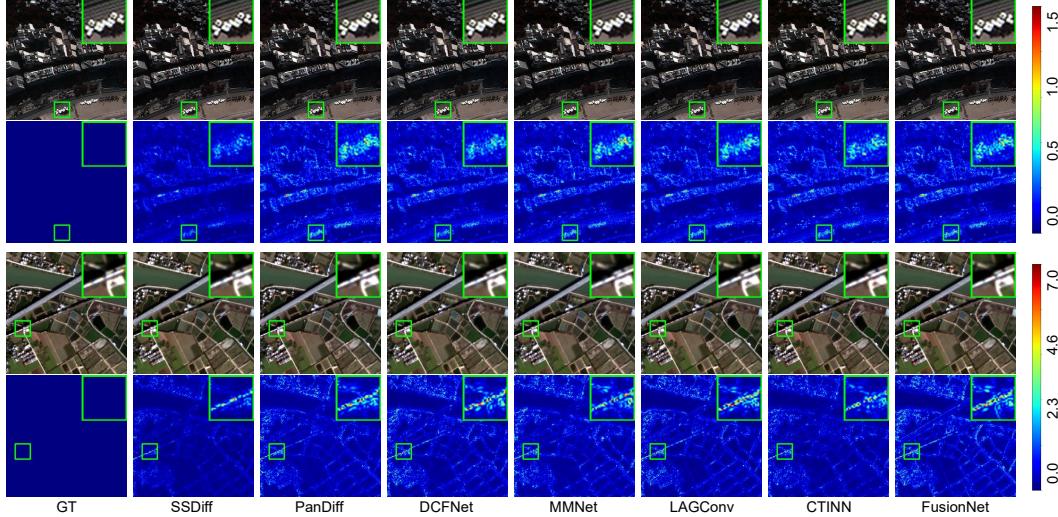


Figure 5: Visual comparisons on a reduced-resolution WorldView-3 and GaoFen-2 case. The first two rows are the results of WV3, and the last two rows are the results of GF2. The first and third rows are the predicted HrMSI for each method, and the second and fourth rows are the error maps of the predicted HRMSI versus ground truth (GT) for each one.

high-frequency components exhibits distinct dynamic variation. Considering the above phenomenon, we design a frequency modulation inter-branch module. Specifically, we utilize a Fourier filter to extract the high-frequency information of the feature map \mathbf{x}_{spa} obtained from the spatial branch, following:

$$\mathcal{F}'(\mathbf{x}_{spa}) = \text{FFT}(\mathbf{x}_{spa}) \odot \alpha, \quad (11)$$

$$\mathbf{x}'_{spa} = \text{IFFT}(\mathcal{F}'(\mathbf{x}_{spa})), \quad (12)$$

where **FFT** and **IFFT** are Fourier transform and inverse Fourier transform. \odot denotes element-wise multiplication, and α is a Fourier mask [27]. Furthermore, we find in experiments that directly injecting high-frequency information into spectral branches will cause the frequency information imbalance. To solve this problem, half of the channels of feature \mathbf{x}_{spec} of the spectral branch are multiplied by a constant. For the three different channel numbers in the model, ranging from low to high, we set this constant to 1.2, 1.4, and 1.6, respectively.

3.4 LoRA-like Branch-wise Alternative Fine-tuning

During the model training process, it is crucial to carefully maintain a balance between model underfitting and overfitting. Our SSDiff can hierarchically and discriminatively extract more features. However, achieving a simultaneous balance condition for both spatial and spectral branches during unified training is undoubtedly challenging. Therefore, this paper approaches the LoRA-like alternate fine-tuning of each branch as a feasible solution. As shown in Fig. 1, LoRA methods fine-tune the output of the model by updating the parameters on the fully connected layer weights. Compared with LoRA, the proposed alternating projection method is also a low-rank matrix decomposition. In Fig. 1 (b), the difference is that we can have the backpropagation process and control the gradient of the projection process to achieve alternate fine-tuning in the proposed APFM. In practice, taking the fine-tuning of Eq. (8) (\mathbf{T}^{spa}) as an example to update the spectral branch, we can detach the gradient propagation at \mathbf{T}^{spa} , preventing parameter updates in the spatial branch. In this case, gradients only propagate through the path shown in Eq. (8) (\mathbf{T}^{spe}), which means only the parameters of the spectral branch are updated. Similarly, when fine-tuning the spatial branch, detaching the gradient propagation from the computation graph at \mathbf{T}^{spe} can prevent parameter updates in the spectral branch. A more intuitive fine-tuning process is provided in the supplementary material.

Table 1: Result on the WV3 (the first thirteen rows) and GF2 (the last thirteen rows) reduced-resolution and full-resolution datasets. The best results are highlighted in bold and the second best results are underlined.

| Method | Reduced resolution | | | | Full resolution | | |
|-----------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| | SAM(\pm std) | ERGAS(\pm std) | $Q2^n$ (\pm std) | SCC(\pm std) | D_λ (\pm std) | D_s (\pm std) | HQNR(\pm std) |
| BDSD-PC [31] | 5.4675 \pm 1.7185 | 4.6549 \pm 1.4667 | 0.8117 \pm 0.1063 | 0.9049 \pm 0.0419 | 0.0625 \pm 0.0235 | 0.0730 \pm 0.0356 | 0.8698 \pm 0.0531 |
| MTF-GLP-FS [33] | 5.3233 \pm 1.6548 | 4.6452 \pm 1.4441 | 0.8177 \pm 0.1014 | 0.8984 \pm 0.0466 | <u>0.0206\pm0.0082</u> | 0.0630 \pm 0.0284 | 0.9180 \pm 0.0346 |
| BT-H [1] | 4.8985 \pm 1.3028 | 4.5150 \pm 1.3315 | 0.8182 \pm 0.1019 | 0.9240 \pm 0.0243 | 0.0574 \pm 0.0232 | 0.0810 \pm 0.0374 | 0.8670 \pm 0.0540 |
| PNN [20] | 3.6798 \pm 0.7625 | 2.6819 \pm 0.6475 | 0.8929 \pm 0.0923 | 0.9761 \pm 0.0075 | 0.0213 \pm 0.0080 | 0.0428 \pm 0.0147 | 0.9369 \pm 0.0212 |
| DiCNN [12] | 3.5929 \pm 0.7623 | 2.6733 \pm 0.6627 | 0.9004 \pm 0.0871 | 0.9763 \pm 0.0072 | 0.0362 \pm 0.0111 | 0.0462 \pm 0.0175 | 0.9195 \pm 0.0258 |
| MSDCNN [36] | 3.7773 \pm 0.8032 | 2.7608 \pm 0.6884 | 0.8900 \pm 0.0900 | 0.9741 \pm 0.0076 | 0.0230 \pm 0.0091 | 0.0467 \pm 0.0199 | 0.9316 \pm 0.0271 |
| FusionNet [5] | 3.3252 \pm 0.6978 | 2.4666 \pm 0.6446 | 0.9044 \pm 0.0904 | 0.9807 \pm 0.0069 | 0.0239 \pm 0.0090 | 0.0364 \pm 0.0137 | <u>0.9406\pm0.0197</u> |
| CTINN [48] | 3.2523 \pm 0.6436 | 2.3936 \pm 0.5194 | 0.9056 \pm 0.0840 | 0.9826 \pm 0.0046 | 0.0550 \pm 0.0288 | 0.0679 \pm 0.0312 | 0.8815 \pm 0.0488 |
| LAGConv [15] | 3.1042 \pm 0.5585 | 2.2999 \pm 0.6128 | 0.9098 \pm 0.0907 | 0.9838 \pm 0.0068 | 0.0368 \pm 0.0148 | 0.0418 \pm 0.0152 | 0.9230 \pm 0.0247 |
| MMNet [49] | 3.0844 \pm 0.6398 | 2.3428 \pm 0.6260 | <u>0.9155\pm0.0855</u> | 0.9829 \pm 0.0056 | 0.0540 \pm 0.0232 | <u>0.0336\pm0.0115</u> | 0.9143 \pm 0.0281 |
| DCFNet [39] | 3.0264 \pm 0.7397 | <u>2.1588\pm0.4563</u> | 0.9051 \pm 0.0881 | <u>0.9861\pm0.0038</u> | 0.0781 \pm 0.0812 | 0.0508 \pm 0.0342 | 0.8771 \pm 0.1005 |
| PanDiff [21] | 3.2968 \pm 0.6010 | 2.4667 \pm 0.5837 | 0.8980 \pm 0.0880 | 0.9800 \pm 0.0063 | 0.0273 \pm 0.0123 | 0.0542 \pm 0.0264 | 0.9203 \pm 0.0360 |
| SSDiff (ours) | 2.8429\pm0.5284 | 2.1059\pm0.4560 | 0.9156\pm0.0841 | 0.9867\pm0.0038 | 0.0132\pm0.0049 | 0.0307\pm0.0029 | 0.9565\pm0.0057 |
| BDSD-PC [31] | 1.7110 \pm 0.3210 | 1.7025 \pm 0.4056 | 0.9932 \pm 0.0308 | 0.9448 \pm 0.0166 | 0.0759 \pm 0.0301 | 0.1548 \pm 0.0280 | 0.7812 \pm 0.0409 |
| MTF-GLP-FS [33] | 1.6757 \pm 0.3457 | 1.6023 \pm 0.3545 | 0.8914 \pm 0.0256 | 0.9390 \pm 0.0197 | 0.0336 \pm 0.0129 | 0.1404 \pm 0.0277 | 0.8309 \pm 0.0334 |
| BT-H [1] | 1.6810 \pm 0.3168 | 1.5524 \pm 0.3642 | 0.9089 \pm 0.0292 | 0.9508 \pm 0.0150 | 0.0602 \pm 0.0252 | 0.1313 \pm 0.0193 | 0.8165 \pm 0.0305 |
| PNN [20] | 1.0477 \pm 0.2264 | 1.0572 \pm 0.2355 | 0.9604 \pm 0.0100 | 0.9772 \pm 0.0054 | 0.0367 \pm 0.0291 | 0.0943 \pm 0.0224 | 0.8726 \pm 0.0373 |
| DiCNN [12] | 1.0525 \pm 0.2310 | 1.0812 \pm 0.2510 | 0.9594 \pm 0.0101 | 0.9771 \pm 0.0058 | 0.0413 \pm 0.0128 | 0.0992 \pm 0.0131 | 0.8636 \pm 0.0165 |
| MSDCNN [36] | 1.0472 \pm 0.2210 | 1.0413 \pm 0.2309 | 0.9612 \pm 0.0108 | 0.9782 \pm 0.0050 | 0.0269 \pm 0.0131 | 0.0730 \pm 0.0093 | 0.9020 \pm 0.0128 |
| FusionNet [5] | 0.9735 \pm 0.2117 | 0.9878 \pm 0.2222 | 0.9641 \pm 0.0093 | 0.9806 \pm 0.0049 | 0.0400 \pm 0.0126 | 0.1013 \pm 0.0134 | 0.8628 \pm 0.0184 |
| CTINN [48] | 0.8251 \pm 0.1386 | 0.6995 \pm 0.1068 | 0.9772 \pm 0.0117 | 0.9803 \pm 0.0015 | 0.0586 \pm 0.0260 | 0.1096 \pm 0.0149 | 0.8381 \pm 0.0237 |
| LAGConv [15] | 0.7859 \pm 0.1478 | <u>0.6869\pm0.1125</u> | <u>0.9804\pm0.0085</u> | <u>0.9906\pm0.0019</u> | 0.0324 \pm 0.0130 | 0.0792 \pm 0.0136 | 0.8910 \pm 0.0204 |
| MMNet [49] | 0.9929 \pm 0.1411 | 0.8117 \pm 0.1185 | 0.9690 \pm 0.0204 | 0.9859 \pm 0.0024 | 0.0428 \pm 0.0300 | 0.1033 \pm 0.0129 | 0.8583 \pm 0.0269 |
| DCFNet [39] | 0.8896 \pm 0.1577 | 0.8061 \pm 0.1369 | 0.9727 \pm 0.0100 | 0.9853 \pm 0.0024 | <u>0.0234\pm0.0116</u> | <u>0.0659\pm0.0096</u> | 0.9122 \pm 0.0119 |
| PanDiff [21] | 0.8881 \pm 0.1197 | 0.7461 \pm 0.1032 | 0.9792 \pm 0.0097 | 0.9887 \pm 0.0020 | 0.0265 \pm 0.0195 | 0.0729 \pm 0.0103 | 0.9025 \pm 0.0209 |
| SSDiff (ours) | 0.6694\pm0.1244 | 0.6038\pm0.1080 | 0.9836\pm0.0074 | 0.9915\pm0.0017 | 0.0164\pm0.0093 | 0.0267\pm0.0071 | 0.9573\pm0.0100 |
| Ideal value | 0 | 0 | 1 | 1 | 0 | 0 | 1 |

4 Experiments

4.1 Experimental Results

Results on WorldView-3. On the WorldView-3 dataset, we evaluate the performance of SSDiff using 20 test images. The results for both reduced-resolution and full-resolution are presented in Table 1. The running time of a single picture during the sampling process is 7.417 seconds under 10 timesteps. We compared our method with three traditional methods and some SOTA DL-based methods. To illustrate the performance of each method more clearly, we presented the fusion result images and error maps of some of these methods in Fig. 5, and zoomed in on a specific location. On average, our method achieves SOTA performance on the reduced dataset, with our SSDiff reaching 2.84 (SAM) and 2.10 (ERGAS) metrics, outperforming all DL-based methods. The error map indicates that the images sampled by SSDiff are closer to the ground truth (GT). SSDiff achieves SOTA performance in obtaining full-resolution images on the WV3 dataset. The HQNR score close to 1 indicates better fusion quality of the full-resolution images. The obtained results demonstrate that our SSDiff can fuse HrMSI, reducing spatial and spectral distortions, thereby proving its excellent generalization ability at full resolution.

Results on GaoFen-2. On the GaoFen-2 reduced dataset, we tested our SSDiff on 20 test images, as shown in Table 1. Our SSDiff achieves SOTA performance. From the error maps in Fig. 5, we can observe that there are still significant differences between traditional fusion methods and DL-based fusion methods. These experiments demonstrate that, compared to other DL-based methods used for comparison in the experiments, the proposed SSDiff exhibits superior spatial performance and effectively preserves spectral information.

4.2 Ablation Study

Effectiveness of Decoupling Branches. To investigate the effectiveness of the spatial and spectral branch design, we perform ablation experiments by training the diffusion model under the following

conditions: V1) Coupling inputs, and training only with the spatial branch. V2) Coupling inputs and training only with the spectral branch. V3) Spatial and spectral branch structure and decoupling inputs, only the outputs of the two branches are concatenated without any inter-branch information interaction. V4) Spatial and spectral branch structure and decoupling inputs, replace each APFM with an additional operation, i.e., the way of LoRA. V5) Spatial and spectral branch structure and decoupling inputs, replace each APFM with a multiplication operation. V6) Our method. The results on the WV3 reduced dataset are reported in Table 2. Training solely with a single branch significantly reduces the quality of the HrMSI. It can be seen that SSDiff is more suitable for pansharpening tasks than the LoRA way, multiplication, and concatenating way.

Table 2: Ablation study on 20 reduced-resolution samples from WV3 dataset without fine-tuning.

| Method | SAM(\pm std) | ERGAS(\pm std) | $Q2^n$ (\pm std) | SCC(\pm std) | Params |
|--------|---------------------------------------|---------------------------------------|---------------------------------------|---------------------------------------|--------|
| V1 | 3.3612 ± 0.6497 | 2.5633 ± 0.6249 | 0.8960 ± 0.1031 | 0.9816 ± 0.0070 | 1100K |
| V2 | 3.0598 ± 0.5560 | 2.2638 ± 0.5663 | 0.9097 ± 0.0947 | 0.9847 ± 0.0062 | 654K |
| V3 | 3.4040 ± 0.6088 | 2.5564 ± 0.6737 | 0.9023 ± 0.0954 | 0.9796 ± 0.0078 | 1420K |
| V4 | 3.3245 ± 0.5342 | 2.4371 ± 0.5841 | 0.9089 ± 0.0878 | 0.9830 ± 0.0069 | 1420K |
| V5 | 3.1958 ± 0.5727 | 2.3962 ± 0.5688 | 0.9035 ± 0.0962 | 0.9834 ± 0.0067 | 1420K |
| V6 | 2.8646 ± 0.5241 | 2.1217 ± 0.4671 | 0.9125 ± 0.0874 | 0.9863 ± 0.0040 | 1420K |

Frequency Modulation Inter-branch Module. To validate the effectiveness of the FMIM, we remove FMIM from our model and train the diffusion model to converge the WV3 dataset. The results are shown in Table 3. Without using FMIM, the model’s performance on the SAM/ERGAS/Q8 indicators decreased by approximately 3.1%/2.8%/1%, respectively. This demonstrates that utilizing FMIM for frequency transfer can effectively improve model performance.

Table 3: Ablation study on 20 reduced-resolution samples from WV3 dataset without fine-tuning.

| FMIM | SAM(\pm std) | ERGAS(\pm std) | $Q2^n$ (\pm std) | SCC(\pm std) |
|------|---------------------------------------|---------------------------------------|---------------------------------------|---------------------------------------|
| ✗ | 2.9798 ± 0.6060 | 2.1978 ± 0.5399 | 0.9153 ± 0.0868 | 0.9855 ± 0.0053 |
| ✓ | 2.8646 ± 0.5241 | 2.1217 ± 0.4671 | 0.9125 ± 0.0874 | 0.9863 ± 0.0040 |

4.3 Discussion

Generalization. To test the generalization ability of DL-based methods, we evaluated models trained on the WV3 dataset using 20 reduced resolutions from the WorldView-2 dataset. The quantitative evaluation results, as reported in Table 4, demonstrate that the SSDiff method achieves the best results across all four evaluation metrics. This indicates that our approach possesses a powerful generalization ability.

Table 4: Generalization of DL-based methods on WV2 dataset.

| Method | SAM(\pm std) | ERGAS(\pm std) | $Q2^n$ (\pm std) | SCC(\pm std) |
|---------------|---------------------------------------|---------------------------------------|---------------------------------------|---------------------------------------|
| PNN | 7.1158 ± 1.6812 | 5.6152 ± 0.9431 | 0.7619 ± 0.0928 | 0.8782 ± 0.0175 |
| DiCNN | 6.9216 ± 0.7898 | 6.2507 ± 0.5745 | 0.7205 ± 0.0746 | 0.8552 ± 0.0289 |
| MSDCNN | 6.0064 ± 0.6377 | 4.7438 ± 0.4939 | 0.8241 ± 0.0799 | 0.8972 ± 0.0109 |
| FusionNet | 6.4257 ± 0.8602 | 5.1363 ± 0.5151 | 0.7961 ± 0.0737 | 0.8746 ± 0.0134 |
| CTINN | 6.4103 ± 0.5953 | 4.6435 ± 0.3792 | 0.8172 ± 0.0873 | 0.9147 ± 0.0102 |
| LAGConv | 6.9545 ± 0.4739 | 5.3262 ± 0.3185 | 0.8054 ± 0.0837 | 0.9125 ± 0.0101 |
| MMNet | 6.6109 ± 0.3209 | 5.2213 ± 0.2133 | 0.8143 ± 0.0790 | 0.9136 ± 0.0201 |
| DCFNet | 5.6194 ± 0.6039 | 4.4887 ± 0.3764 | 0.8292 ± 0.0815 | 0.9154 ± 0.0083 |
| SSDiff (ours) | 5.0647 ± 0.5634 | 3.9885 ± 0.4297 | 0.8577 ± 0.0782 | 0.9335 ± 0.0055 |

Training SSDiff. To address the issue of insufficient local parameter training in the dual branches model, we design the L-BAF method to fine-tune the model. Taking the experiments on the reduced WV3 dataset, we first train the model without fine-tuning until convergence and perform branch-wise fine-tuning, which includes: 1) Full parameter fine-tuning without L-BAF. 2) Only fine-tuning the spatial branch parameters with the spectral branch parameters fixed. 3) Only fine-tuning the spectral

branch parameters with the spatial parameters fixed. 4) Alternating fine-tuning spectral branch and spatial branch. The quantitative results are shown in Table 5, and all three branch-wise fine-tuning methods lead to improved testing performance. The results of full fine-tuning show a decrease in some metrics, indicating that the method suffers from overfitting. The alternating fine-tuning approach showed the most significant performance improvement. This demonstrates the effectiveness of the L-BAF.

Table 5: Fine-tune SSDiff. \mathcal{S} and \mathcal{F} denote fine-tuning of spatial and spectral branches, respectively.

| \mathcal{S} | \mathcal{F} | SAM(\pm std) | ERGAS(\pm std) | Q2 ⁿ (\pm std) | SCC(\pm std) |
|---------------|---------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| - | - | 2.8646 \pm 0.5241 | 2.1217 \pm 0.4671 | 0.9125 \pm 0.0874 | 0.9863 \pm 0.0040 |
| \times | \times | 2.8681 \pm 0.5837 | 2.1302 \pm 0.5235 | 0.9206\pm0.0850 | 0.9868\pm0.0048 |
| \checkmark | \times | 2.8545 \pm 0.5244 | 2.1138 \pm 0.4658 | 0.9143 \pm 0.0857 | 0.9864 \pm 0.0040 |
| \times | \checkmark | 2.8460 \pm 0.5232 | 2.1132 \pm 0.4671 | 0.9152 \pm 0.0849 | 0.9864 \pm 0.0041 |
| \checkmark | \checkmark | 2.8429\pm0.5284 | 2.1059\pm0.4560 | 0.9156 \pm 0.0841 | 0.9867 \pm 0.0038 |

5 Conclusion

In this paper, we propose a spatial-spectral integrated diffusion model for remote sensing pansharpening, named SSDiff. We design a spatial-spectral integrated model architecture, which utilizes spatial and spectral branches to learn spatial details and spectral features separately. By introducing vector projection, the spatial and spectral components in the subspace decomposition are further specified in the proposed APFM. Then, the self-attention mechanism is naturally generalized to the APFM. Furthermore, we propose an FMIM to modulate the frequency distribution between branches. Finally, the two branches of SSDiff can capture discriminating features. It is interesting that, when utilizing the proposed L-BAF method in the APFM, the two branches can be updated alternately, and then SSDiff produces more satisfactory results. We compare our SSDiff with several SOTA pansharpening methods on the WorldView-3, QuickBird, GaoFen-2, and WorldView-2 datasets. The results demonstrate the superiority of SSDiff both visually and quantitatively.

6 Acknowledgement

This work is supported by the National Natural Science Foundation of China under Grants 12271083, 12171072 and Natural Science Foundation of Sichuan Province under Grants 2023NSFSC1341.

References

- [1] Bruno Aiazzi, L Alparone, Stefano Baronti, Andrea Garzelli, and Massimo Selva. Mtf-tailored multiscale fusion of high-resolution ms and pan imagery. *Photogrammetric Engineering & Remote Sensing*, 72(5):591–596, 2006.
- [2] Bruno Aiazzi, Luciano Alparone, Stefano Baronti, and Andrea Garzelli. 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, 2002.
- [3] Alberto Arienzo, Gemine Vivone, Andrea Garzelli, Luciano Alparone, and Jocelyn Chanussot. Full-resolution quality assessment of pansharpening: Theoretical and hands-on approaches. *IEEE Geoscience and Remote Sensing Magazine*, 10(3):168–201, 2022.
- [4] Zihan Cao, Shiqi Cao, Liang-Jian Deng, Xiao Wu, Junming Hou, and Gemine Vivone. Diffusion model with disentangled modulations for sharpening multispectral and hyperspectral images. *Information Fusion*, 104:102158, 11 2023.
- [5] Liang-Jian Deng, Gemine Vivone, Cheng Jin, and Jocelyn Chanussot. Detail injection-based deep convolutional neural networks for pansharpening. *IEEE Transactions on Geoscience and Remote Sensing*, 59(8):6995–7010, 2020.
- [6] Shangqi Deng, Liang-Jian Deng, Xiao Wu, Ran Ran, and Rui Wen. Bidirectional dilation transformer for multispectral and hyperspectral image fusion. In *Proc. 32nd Int. Joint Conf. Artif. Intell.*, pages 3633–3641, 2023.

- [7] Renwei Dian and Shutao Li. Hyperspectral image super-resolution via subspace-based low tensor multi-rank regularization. *IEEE Transactions on Image Processing*, 28(10):5135–5146, 2019.
- [8] Junkai Fan, Jiangwei Weng, Kun Wang, Yijun Yang, Jianjun Qian, Jun Li, and Jian Yang. Driving-video dehazing with non-aligned regularization for safety assistance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26109–26119, 2024.
- [9] Sicheng Gao, Xuhui Liu, Bohan Zeng, Sheng Xu, Yanjing Li, Xiaoyan Luo, Jianzhuang Liu, Xiantong Zhen, and Baochang Zhang. Implicit diffusion models for continuous super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10021–10030, 2023.
- [10] Andrea Garzelli and Filippo Nencini. Hypercomplex quality assessment of multi/hyperspectral images. *IEEE Geoscience and Remote Sensing Letters*, 6(4):662–665, 2009.
- [11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [12] Lin He, Yizhou Rao, Jun Li, Jocelyn Chanussot, Antonio Plaza, Jiawei Zhu, and Bo Li. Pansharpening via detail injection based convolutional neural networks. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(4):1188–1204, 2019.
- [13] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.
- [14] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.
- [15] Zi-Rong Jin, Tian-Jing Zhang, Tai-Xiang Jiang, Gemine Vivone, and Liang-Jian Deng. Lagconv: Local-context adaptive convolution kernels with global harmonic bias for pansharpening. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 1113–1121, 2022.
- [16] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6007–6017, 2023.
- [17] P Kwarteng and A Chavez. Extracting spectral contrast in landsat thematic mapper image data using selective principal component analysis. *Photogramm. Eng. Remote Sens*, 55(1):339–348, 1989.
- [18] Yixun Liang, Ping Zhang, Yang Mei, and Tingqi Wang. Pmacnet: Parallel multiscale attention constraint network for pan-sharpening. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2022.
- [19] Yu-Jie Liang, Zihan Cao, Liang-Jian Deng, and Xiao Wu. Fourier-enhanced implicit neural fusion network for multispectral and hyperspectral image fusion. *arXiv preprint arXiv:2404.15174*, 2024.
- [20] Giuseppe Masi, Davide Cozzolino, Luisa Verdoliva, and Giuseppe Scarpa. Pansharpening by convolutional neural networks. *Remote Sensing*, 8(7):594, 2016.
- [21] Qingyan Meng, Wenxu Shi, Sijia Li, and Linlin Zhang. Pandiff: A novel pansharpening method based on denoising diffusion probabilistic model. *IEEE Transactions on Geoscience and Remote Sensing*, 2023.
- [22] Xiangchao Meng, Jie Li, Huanfeng Shen, Liangpei Zhang, and Hongyan Zhang. Pan sharpening with a guided filter based on three-layer decomposition. *Sensors*, 16(7):1068, 2016.

- [23] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021.
- [24] Xavier Otazu, María González-Audícana, Octavi Fors, and Jorge Núñez. Introduction of sensor spectral response into image fusion methods. application to wavelet-based methods. *IEEE Transactions on Geoscience and Remote Sensing*, 43(10):2376–2385, 2005.
- [25] Siran Peng, Chenhao Guo, Xiao Wu, and Liang-Jian Deng. U2net: A general framework with spatial-spectral-integrated double u-net for image fusion. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 3219–3227, 2023.
- [26] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22500–22510, 2023.
- [27] Chenyang Si, Ziqi Huang, Yuming Jiang, and Ziwei Liu. Freeu: Free lunch in diffusion u-net. *arXiv preprint arXiv:2309.11497*, 2023.
- [28] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.
- [29] Gilbert Strang. *Introduction to linear algebra*. SIAM, 2022.
- [30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [31] Gemine Vivone. Robust band-dependent spatial-detail approaches for panchromatic sharpening. *IEEE transactions on Geoscience and Remote Sensing*, 57(9):6421–6433, 2019.
- [32] Gemine Vivone, Rocco Restaino, and Jocelyn Chanussot. A regression-based high-pass modulation pansharpening approach. *IEEE Transactions on Geoscience and Remote Sensing*, 56(2):984–996, 2017.
- [33] Gemine Vivone, Rocco Restaino, and Jocelyn Chanussot. Full scale regression-based injection coefficients for panchromatic sharpening. *IEEE Transactions on Image Processing*, 27(7):3418–3431, 2018.
- [34] Lucien Wald. *Data fusion: definitions and architectures: fusion of images of different spatial resolutions*. Presses des MINES, 2002.
- [35] Lucien Wald, Thierry Ranchin, and Marc Mangolini. Fusion of satellite images of different spatial resolutions: Assessing the quality of resulting images. *Photogrammetric Engineering and Remote Sensing*, 63(6):691–699, 1997.
- [36] Yancong Wei, Qiangqiang Yuan, Xiangchao Meng, Huanfeng Shen, Liangpei Zhang, and Michael Ng. Multi-scale-and-depth convolutional neural network for remote sensed imagery pan-sharpening. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 3413–3416. IEEE, 2017.
- [37] Chen Wu, Bo Du, Xiaohui Cui, and Liangpei Zhang. A post-classification change detection method based on iterative slow feature analysis and bayesian soft fusion. *Remote Sensing of Environment*, 199:241–255, 2017.
- [38] Hongtao Wu, Yijun Yang, Haoyu Chen, Jingjing Ren, and Lei Zhu. Mask-guided progressive network for joint raindrop and rain streak removal in videos. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 7216–7225, 2023.
- [39] Xiao Wu, Ting-Zhu Huang, Liang-Jian Deng, and Tian-Jing Zhang. Dynamic cross feature fusion for remote sensing pansharpening. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14687–14696, 2021.

- [40] Zhong-Cheng Wu, Ting-Zhu Huang, Liang-Jian Deng, Jie Huang, Jocelyn Chanussot, and Gemine Vivone. Lrtcfpan: Low-rank tensor completion based framework for pansharpening. *IEEE Transactions on Image Processing*, 32:1640–1655, 2023.
- [41] Zhong-Cheng Wu, Ting-Zhu Huang, Liang-Jian Deng, and Gemine Vivone. A framelet sparse reconstruction method for pansharpening with guaranteed convergence. *Inverse Problems and Imaging*, pages 0–0, 2023.
- [42] Yijun Yang, Angelica I Aviles-Rivero, Huazhu Fu, Ye Liu, Weiming Wang, and Lei Zhu. Video adverse-weather-component suppression network via weather messenger and adversarial backpropagation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13200–13210, 2023.
- [43] Yijun Yang, Huazhu Fu, Angelica I Aviles-Rivero, Carola-Bibiane Schönlieb, and Lei Zhu. Diffmic: Dual-guidance diffusion network for medical image classification. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 95–105. Springer, 2023.
- [44] Yijun Yang, Hongtao Wu, Angelica I Aviles-Rivero, Yulun Zhang, Jing Qin, and Lei Zhu. Genuine knowledge from practice: Diffusion test-time adaptation for video adverse weather removal. *arXiv preprint arXiv:2403.07684*, 2024.
- [45] Xiaohui Yuan, Jianfang Shi, and Lichuan Gu. A review of deep learning methods for semantic segmentation of remote sensing imagery. *Expert Systems with Applications*, 169:114417, 2021.
- [46] Roberta H Yuhas, Alexander FH Goetz, and Joe W Boardman. Discrimination among semi-arid landscape endmembers using the spectral angle mapper (sam) algorithm. In *JPL, Summaries of the Third Annual JPL Airborne Geoscience Workshop. Volume 1: AVIRIS Workshop*, 1992.
- [47] Jie Zhou, Daniel L Civco, and John A Silander. A wavelet transform method to merge landsat tm and spot panchromatic data. *International Journal of Remote Sensing*, 19(4):743–757, 1998.
- [48] Man Zhou, Jie Huang, Yanchi Fang, Xueyang Fu, and Aiping Liu. Pan-sharpening with customized transformer and invertible neural network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 3553–3561, 2022.
- [49] Man Zhou, Keyu Yan, Jinshan Pan, Wenqi Ren, Qi Xie, and Xiangyong Cao. Memory-augmented deep unfolding network for guided image super-resolution. *International Journal of Computer Vision*, 131(1):215–242, 2023.

A Appendix / supplemental material

In the supplementary material, we first introduce the background knowledge, i.e., denoising diffusion probabilistic models, and then illustrate the process of model training and LoRA-like branch-wise alternative fine-tuning, the limitations, and the broader impact of our method. After that, we provide more details for experiments on the pansharpening task, i.e., Implementation Details, Datasets, Benchmarks, and Quality Metrics. Finally, we display more qualitative evaluation results on the QuickBird and GaoFen-2 datasets.

B Background

B.1 Denoising Diffusion Probabilistic Models

Denoising diffusion probabilistic models [13] are latent variable models, generating realistic target images progressively from a normal distribution by iterative denoising. The diffusion model contains two steps: forward and reverse processes.

The forward process aims to make the prior data distribution \mathbf{x}_0 noisy by a T step Markov chain that gradually transforms it into an approximate standard normal distribution $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I}_d)$ and d denotes the dimension. One forward step is defined as follows:

$$q(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1 - \beta_t} \mathbf{x}_{t-1}, \beta_t \mathbf{I}), \quad (13)$$

where $\mathcal{N}(\cdot)$ is a Gaussian distribution with the mean of $\sqrt{1 - \beta_t} \mathbf{x}_{t-1}$ and variance of $\beta_t \mathbf{I}$, β_t is a pre-defined variance schedule in time step $t \in [0, T]$. Through the reparameterization trick, we can derive \mathbf{x}_t directly from \mathbf{x}_0 . The following equation gives this derivation:

$$q(\mathbf{x}_t | \mathbf{x}_0) = \sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon, \quad (14)$$

where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ and $\alpha_t = 1 - \beta_t$, $\bar{\alpha}_t = \prod_{i=0}^t \alpha_i$.

The reverse process aims to learn to remove the degradation brought from the forward process and sample the \mathbf{x}_0 from \mathbf{x}_t . To accomplish this objective, we need to learn the distribution of $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)$ using a neural network and perform iterative sampling as follows:

$$p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t)), \quad (15)$$

where μ_θ and Σ_θ are the mean and variance of $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)$, respectively, and θ is the parameters of model.

According to Eq. (15), the mean and variance can be computed, following:

$$\mu_\theta = \frac{1}{\sqrt{\alpha_t}} (\mathbf{x}_t - \frac{\beta_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(\mathbf{x}_t, t)), \quad (16)$$

$$\Sigma_\theta(\mathbf{x}_t, t) = \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t} \beta_t. \quad (17)$$

For sampling from a standard Gaussian noise \mathbf{x}_T to get \mathbf{x}_{T-1} , after performing T -step iterations of sampling as described above, we get the output \mathbf{x}_0 from \mathbf{x}_T .

C LoRA-like Branch-wise Alternative Fine-tuning

In Section 3.4 of the main text. The L-BAF method alternately fine-tunes the spatial and spectral branches based on the proposed APFM. As shown in Fig. 6. When we fine-tune the spectral branch, we freeze the \mathbf{T}_a and \mathbf{T}_b parameters in the spatial branch and the \mathbf{T}_a parameter in the spectral branch. Block gradient propagation to prevent parameter updates for the entire spatial branch. Similarly, when fine-tuning the spatial branch, we freeze the \mathbf{T}_c and \mathbf{T}_d parameters in the spectral branch, as well as the \mathbf{T}_c parameter in the spatial branch. Block gradient propagation to prevent parameter updates for entire spectral branches.

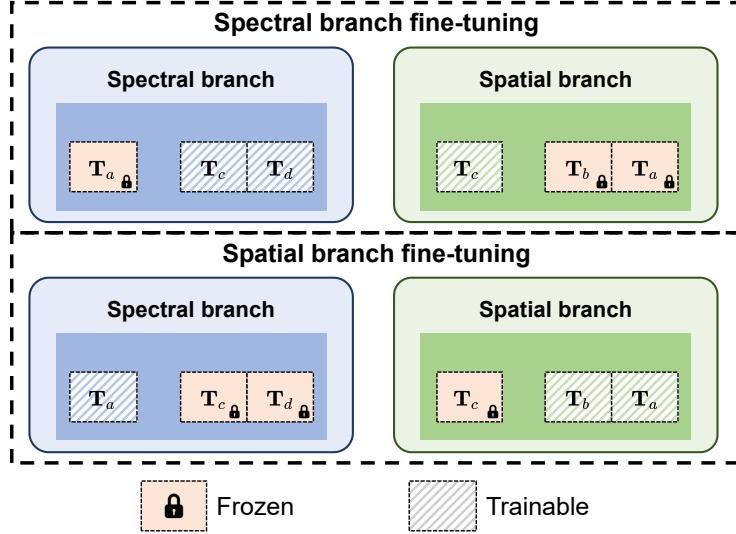


Figure 6: The sketch of the proposed LoRA-like branch-wise alternative fine-tuning process.

Algorithm 1: Training stage of the proposed method.

Data: GT image \mathbf{x}_0 , diffusion model \mathbf{x}_θ with its parameters θ , spectral and spatial branch parameter $\theta_{spe}, \theta_{spa}$, respectively, condition $cond$, timestep t , and denoised objective $\hat{\mathbf{x}}_0$.
Result: Optimized diffusion model \mathbf{x}_θ^* .

```

1 cond ← PAN, LrMSI,  $\mathbf{x}_t$ ;
2 while until convergence do
3    $t \leftarrow \text{Uniform}(0, T)$ ;  $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ ;
4    $\mathbf{x}_t \leftarrow \sqrt{\bar{\alpha}_t}(\mathbf{x}_0 - \text{LrMSI}) + \sqrt{1 - \bar{\alpha}_t}\epsilon$ ;  $\hat{\mathbf{x}}_0 \leftarrow \mathbf{x}_\theta(\mathbf{x}_t, \mathbf{cond}) + \text{LrMSI}$ ;
5   if iteration > 150k then
6     | fine-tune  $\theta_{spe}$  or  $\theta_{spa}$ ; // L-BAF
7   end
8    $\theta \leftarrow \nabla_\theta \mathcal{L}_{simple}(\hat{\mathbf{x}}_0, \mathbf{x}_0)$ .
9 end
```

D Model Training

The detailed training process of SSDiff can be found in Algorithm 1.

E Limitation

First, we evaluate the effectiveness of the proposed SSDiff over the pansharpening task and we will extend our method to other multispectral fusion tasks, such as multispectral and hyperspectral image fusion. Second, the proposed FMIM can adjust the frequency information between the two branches, but it also introduces additional hyperparameters, increasing the difficulty of fine-tuning during training. In terms of cost, the average inference time for a single image is 7.416 seconds under 10 timesteps. The time cost for our approach is higher than other DL-based models, primarily due to the limitation imposed by the large number of sampling steps required in the diffusion model.

F Broader Impact

Pansharpening, as a fundamental preprocessing method which is a key pre-processing technology overcoming the constraints of hardware before using high-resolution multispectral images, has been widely utilized in various applications, including change detection, environmental monitoring, and segmentation. At the same time, pansharpening, being a low-level task, the distortions that occur

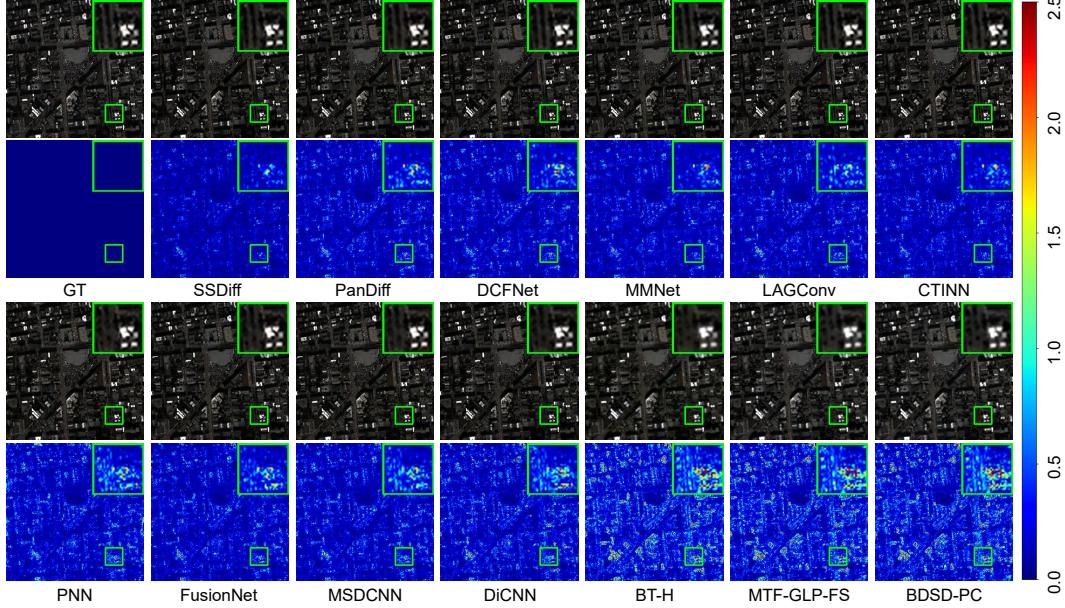


Figure 7: Visual comparisons on a reduced-resolution QuickBird case. The first and third rows are the predicted HrMSI for each method, and the second and fourth rows are the error maps of the predicted HRMS versus ground truth (GT) for each one.

Table 6: Quantitative results on the QuickBird reduced-resolution and full-resolution datasets. Some conventional methods (the first three rows) and DL-based methods are compared. The best results are highlighted in bold and the second best results are underlined.

| Method | Reduced resolution | | | | D_λ (± std) | Full resolution | |
|-----------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|
| | SAM(± std) | ERGAS(± std) | Q4(± std) | SCC(± std) | | D_s (± std) | HQNR(± std) |
| BDSD-PC [31] | 8.2620±2.0497 | 7.5420±0.8138 | 0.8323±0.1013 | 0.9030±0.0181 | 0.1975±0.0334 | 0.1636±0.0483 | 0.6722±0.0577 |
| MTF-GLP-FS [33] | 8.1131±1.9553 | 7.5102±0.7926 | 0.8296±0.0905 | 0.8998±0.0196 | 0.0489±0.0149 | 0.1383±0.0238 | 0.8199±0.0340 |
| BT-H [1] | 7.1943±1.5523 | 7.4008±0.8378 | 0.8326±0.0880 | 0.9156±0.0152 | 0.2300±0.0718 | 0.1648±0.0167 | 0.6434±0.0645 |
| PNN [20] | 5.2054±0.9625 | 4.4722±0.3734 | 0.9180±0.0938 | 0.9711±0.0123 | 0.0569±0.0112 | 0.0624±0.0239 | 0.8844±0.0304 |
| DiCNN [12] | 5.3795±1.0266 | 5.1354±0.4876 | 0.9042±0.0942 | 0.9621±0.0133 | 0.0920±0.0143 | 0.1067±0.0210 | 0.8114±0.0310 |
| MSDCNN [36] | 5.1471±0.9342 | 4.3828±0.3400 | 0.9188±0.0966 | 0.9689±0.0121 | 0.0602±0.0150 | 0.0667±0.0289 | 0.8774±0.0388 |
| FusionNet [5] | 4.9226±0.9077 | 4.1594±0.3212 | 0.9252±0.0902 | 0.9755±0.0104 | 0.0586±0.0189 | 0.0522±0.0088 | 0.8922±0.0219 |
| CTINN [48] | 4.6583±0.7755 | 3.6969±0.2888 | 0.9320±0.0072 | 0.9829±0.0072 | 0.1738±0.0332 | 0.0731±0.0237 | 0.7663±0.0432 |
| LAGConv [15] | 4.5473±0.8296 | 3.8259±0.4196 | 0.9335±0.0878 | 0.9807±0.0091 | 0.0844±0.0238 | 0.0676±0.0136 | 0.8536±0.0178 |
| MMNet [49] | 4.5568±0.7285 | 3.6669±0.3036 | 0.9337±0.0941 | 0.9829±0.0070 | 0.0890±0.0512 | 0.0972±0.0382 | 0.8225±0.0319 |
| DCFNet [39] | 4.5383±0.7397 | 3.8315±0.2915 | 0.9325±0.0903 | 0.9741±0.0101 | 0.0454±0.0147 | 0.1239±0.0269 | 0.8360±0.0158 |
| PanDiff [21] | 4.5754±0.7359 | 3.7422±0.3099 | 0.9345±0.0902 | 0.9818±0.0092 | 0.0587±0.0223 | 0.0642±0.0252 | 0.8813±0.0417 |
| SSDiff (ours) | 4.4640±0.7473 | 3.6320±0.2749 | 0.9346±0.0943 | 0.9829±0.0080 | 0.0314±0.0108 | 0.0360±0.0133 | 0.9338±0.0208 |
| Ideal value | 0 | 0 | 1 | 1 | 0 | 0 | 1 |

during the process of generating high-resolution multispectral images can significantly impact the success rate of subsequent high-level tasks.

G Additional experiment

G.1 Setups

Implementation Details. Our SSDiff is implemented in PyTorch 1.7.0 and Python 3.8.5 using AdamW optimizer with an initial learning rate of 0.001 to minimize \mathcal{L}_{simple} on a Linux operating system with an Intel 12th Gen i7-12700K processor and two NVIDIA GeForce RTX3090 GPUs. For the diffusion denoising model, the initial number of model channels is 32, the diffusion time step used for training in the pansharpening is set to 1000, while the diffusion time step for sampling is set to 10. The exponential moving average (EMA) ratio is set to 0.9999. The total training iterations for the WV3, GF2, and QB datasets are set to 150k, 100k, and 200k iterations, respectively. During the

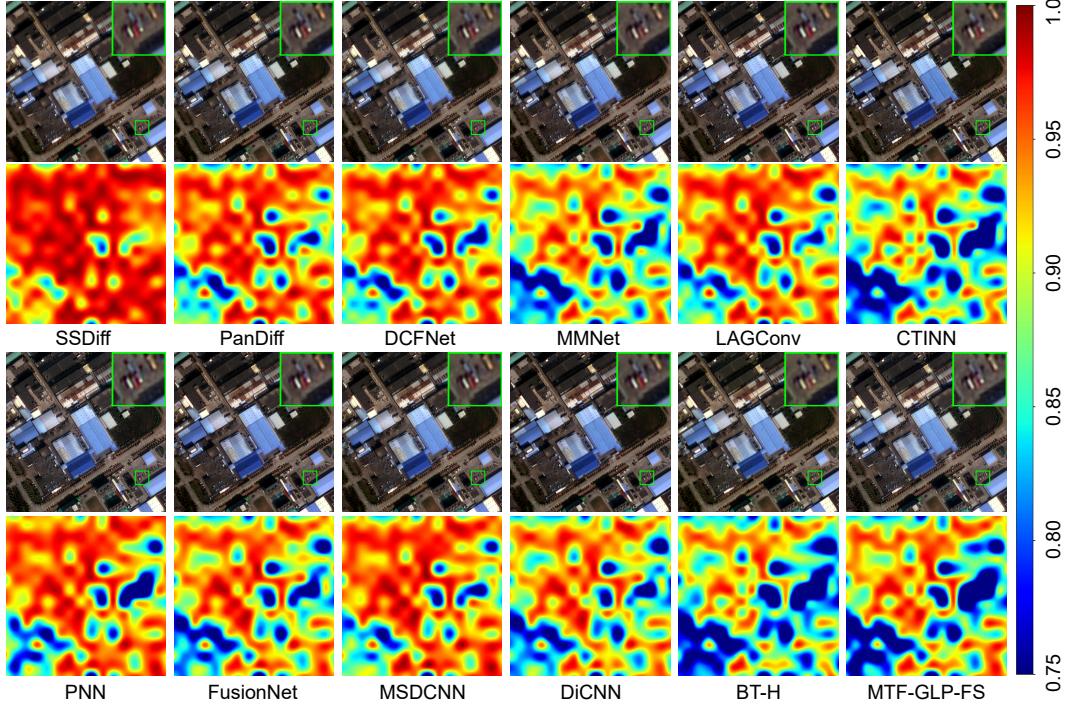


Figure 8: Fused GF2 full-resolution data and their corresponding HQNR map. The high value in the HQNR map means better full-resolution fusion performance.

model fine-tuning, the learning rate is set to 0.0001, and the total fine-tune training iterations are set to 30k.

Datasets. To demonstrate the effectiveness of our SSDiff, we conducted experiments on widely used pansharpening datasets. The Pancollection¹ dataset for pansharpening consists of data from four satellites: WorldView-3 (8 bands), WorldView-2 (8 bands), QuickBird (4 bands), and GaoFen-2 (4 bands).

To better evaluate performance, we simulated reduced-resolution and full-resolution datasets. For reduced datasets, Pancollection follows Wald’s protocol [35] to obtain simulated images with ground truth images. There are three steps involved: 1) Use a modulation transfer-based (MTF) filter to downsample the original PAN and MS images by a factor of 4. Downsampled PAN and MS are used as training PAN and MS images; 2) Treat the original MS image as a ground truth image, i.e. HRMSI; 3) Upsampling the training MS image using interpolation with polynomial kernels [2] and processing it into a LrMSI. When processing the full datasets, the original MS image is considered MS, the upsampled MS image is considered LrMSI, and the original PAN is considered PAN.

Benchmark. To evaluate the performance of our SSDiff, we compared it with various state-of-the-art methods of Pansharpening (on WV3, QB, and GF2 datasets). Specifically, we choose three traditional methods: BDSD-PC [31], MTF-GLP-FS [33], BT-H [1]; as well as nine machine learning-based methods: PNN [20], DiCNN [12], MSDCNN [36], FusionNet [5], CTINN [48], LAGConv [15], MMNet [49], DCFNet [39], and Pandiff [21]. To ensure fairness, we train DL-based methods using the same Nvidia GPU-3090 and PyTorch environment.

Quality Metric. For the reduced data in Pansharpening tasks, we utilize four metrics to evaluate the results on reduced resolution datasets, including the spectral angle mapper (SAM) [46], the erreur relative globale adimensionnelle de synthèse (ERGAS) [34], the universal image quality index ($Q2^n$) [10], and the spatial correlation coefficient (SCC) [47]. As for full-resolution datasets, we apply D_λ , D_s , and hybrid quality with no reference (HQNR) indexes [3] for evaluation.

¹<https://liangjiandeng.github.io/PanCollection.html>.

G.2 Results on QuickBird:

We conduct experiments on the QuickBird reduced dataset and evaluate the performance of SSDiff. Similarly, the reference and non-reference metrics were obtained from 20 randomly selected test images from the QB dataset. The performance comparison is reported in Table 6. Our SSDiff achieves SOTA performance. From the error maps in Fig. 7, we can observe that there are still significant differences between traditional fusion methods and DL-based fusion methods.

G.3 Results on GaoFen-2:

On the GaoFen-2 full-resolution dataset, we tested our SSDiff on 20 test images, Fig. 8 shows the results and HQNR maps, where an HQNR score close to 1 indicates better fusion quality of full-resolution images. The obtained results indicate that our SSDiff has a good generalization of the full-resolution dataset.

Table 7: Efficiency results on the WV3 reduced-resolution datasets.

| Method | SSDiff | PanDiff | DCFNet | MMNet | LAGConv |
|-------------|--------|---------|--------|-------|---------|
| Runtime (s) | 7.417 | 261.410 | 0.548 | 0.348 | 1.381 |

G.4 Efficiency analysis

The diffusion-based method generally has more running time than CNN-based methods due to the multiple timesteps of the diffusion mechanism. The comparison of inference running time shown in Table 7 ensures this point. However, *for a fair comparison* with another diffusion-based method for pansharpening, i.e., PanDiff, our method still gets a significant advantage.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [Yes]

Justification: The abstract and introduction clearly outline the main contributions and scope of the paper, ensuring that the claims made are supported by the theoretical and experimental results presented.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: The paper discusses the limitations in Appendix E, such as the inference overhead of the model and the additional hyperparameters introduced.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory Assumptions and Proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: For Theorem 1 presented in the paper, a complete proof is given in the main text, along with detailed explanations and illustrations.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental Result Reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper provides detailed descriptions of the network architecture and experimental settings and provides links to the publicly available datasets used. The complete code will be published on GitHub upon acceptance for further research and discussion.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The datasets are publicly accessible, and the code will be released soon.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental Setting/Details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: In Appendix G, we present the datasets, implementation details, and the hyperparameters can be found in Sec. 3.3.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment Statistical Significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: The paper does not involve error bars or experiments concerning statistical significance.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments Compute Resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Our experiments were conducted on a workstation equipped with an Intel 12th Gen i7-12700K processor, two NVIDIA RTX 3090 GPUs, and 128GB of memory. The detailed compute resources for the experiments can be found in the implementation details of Appendix. G.1.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code Of Ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: The research adheres to the ethical guidelines set forth by NeurIPS, ensuring responsible conduct in all aspects of the study.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader Impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: Both potential positive societal impacts and negative societal impacts of the work can be found in the implementation details of Appendix. F.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The models compared in the paper have either received author permission or are publicly available. The datasets are public, and original papers are cited meticulously.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The models compared in the paper have either received author permission or are publicly available. The datasets are public, and original papers are cited meticulously.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New Assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: The paper does not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and Research with Human Subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing or research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing or research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing or research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing or research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.