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

Yu Zhong[†,1] , Xiao Wu[†,1]

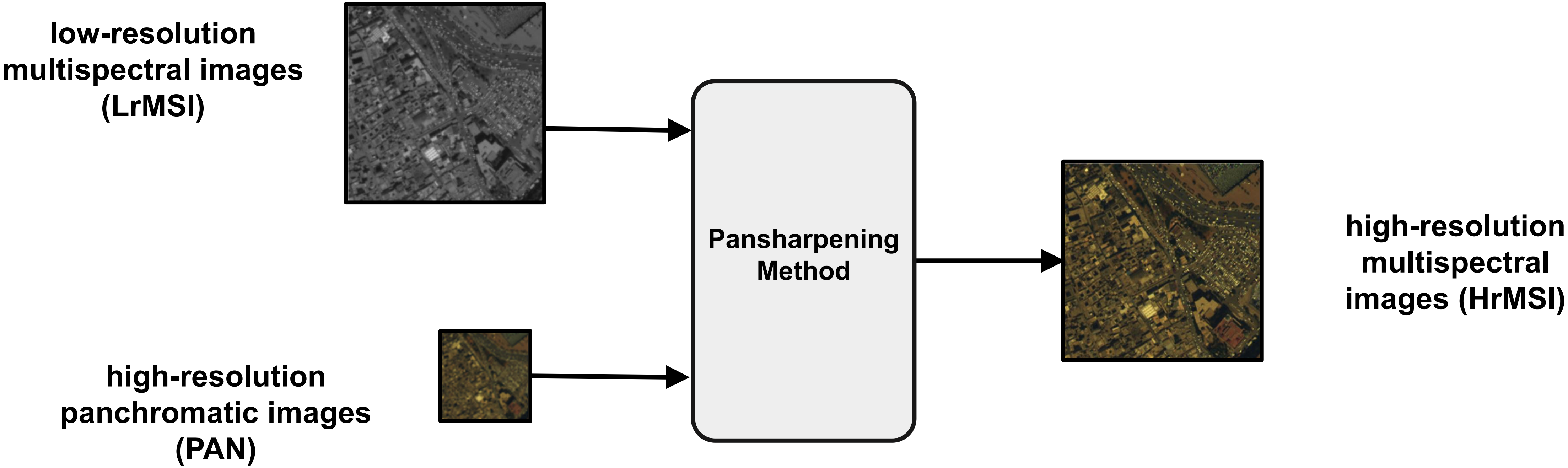
Zihan Cao[1], Hong-Xia Dou[2], Liang-Jian Deng[*,1]

[1]University of Electronic Science and Technology of China
[2]Xihua University

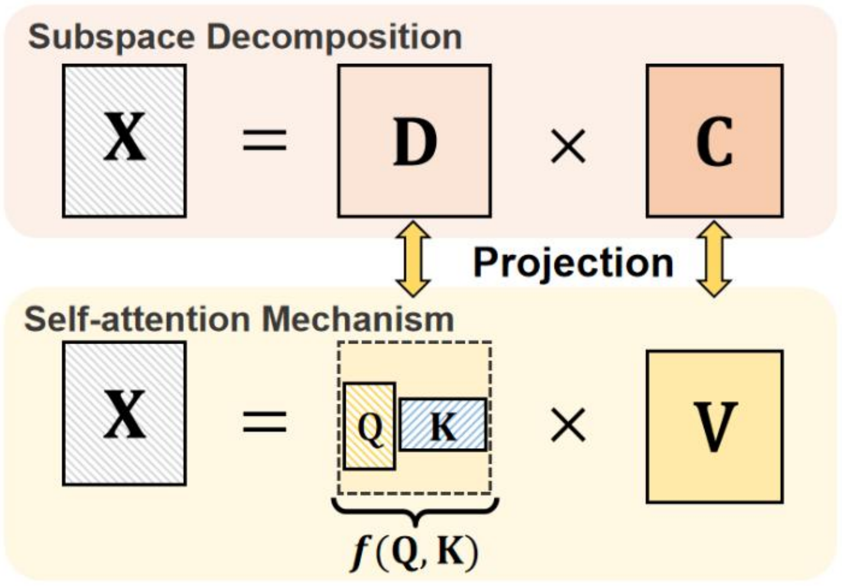
Outline

- Background: Pansharpening
- Motivation and Alternating Projection Fusion
- SSDiff and LoRA-like Fine-tuning
- Experimental Results

Background: Panchromatic and Multispectral Image Fusion (Pansharpening)



Motivation



DDPM-based methods:

- **Existing DDPM-based methods have not yet designed models specifically for the discriminative features required in the pansharpening task.**

Our method:

- 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.

Methodology: Alternating Projection Fusion Method (APFM)

Lemma 1 ([Strang, 2022]). *Assuming that the existing two arbitrary vectors* $\mathbf{a} \in dom\mathbf{U} \in \mathbb{R}^n$ *and* $\mathbf{b} \in dom\mathbf{G} \in \mathbb{R}^n$, *then* $\mathbf{Pb} = \lambda\mathbf{a} = \mathbf{p}$, *we have the following formula:*

$$\mathbf{p} = \frac{\mathbf{a}\mathbf{a}^T}{\mathbf{a}^T\mathbf{a}}\mathbf{b}. \tag{1}$$

where $\mathbf{P}$ *is a projection matrix,* λ *denotes 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:

$$\mathbf{a}^T\mathbf{e} = \mathbf{a}^T(\mathbf{p} - \mathbf{b}) = \mathbf{a}^T(\lambda\mathbf{a} - \mathbf{b}) = 0, \tag{2}$$

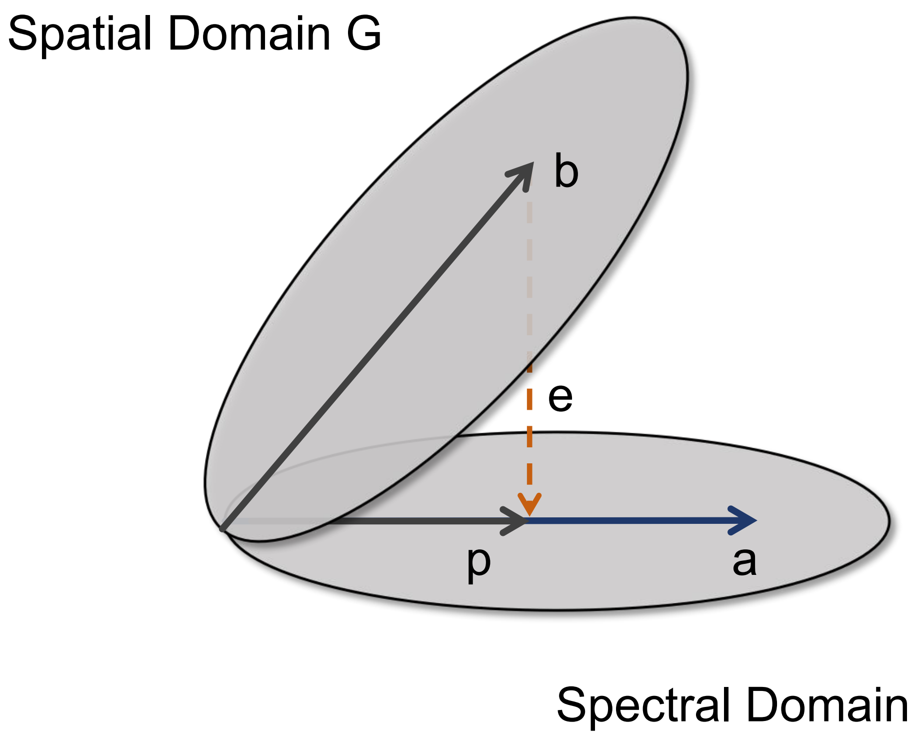
thus, we have

$$\lambda = \frac{\mathbf{a}^T\mathbf{b}}{\mathbf{a}^T\mathbf{a}}. \tag{3}$$

Taking Eq. (3) into $\mathbf{Pb} = \lambda\mathbf{a} = \mathbf{p}$, we have the conclusion:

$$\mathbf{p} = \mathbf{Pb} = \mathbf{a}\lambda = \frac{\mathbf{a}\mathbf{a}^T}{\mathbf{a}^T\mathbf{a}}\mathbf{b}. \tag{4}$$

$\square$

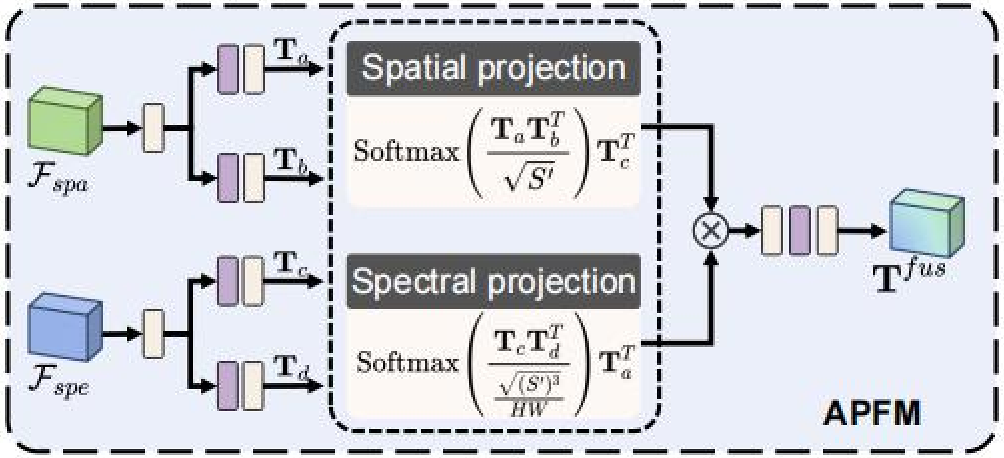


Methodology: Alternating Projection Fusion Method (APFM)

$$Attention(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$$

Proof. According to Lemma 1, we can generalize the self-attention mechanism [Vaswani *et al.*, 2017], where $\mathbf{Q}$ and $\mathbf{K}$ are the features from dom$\mathbf{U}$, and $\mathbf{V}$ is the feature from dom$\mathbf{G}$, respectively. Thus, we have the following form:

$$\text{Softmax}(\frac{\mathbf{QK}^T}{\sqrt{d_k}}) = \frac{\mathbf{aa}^T}{\mathbf{a}^T\mathbf{a}},\ \mathbf{V} = \mathbf{b}. \quad (5)$$

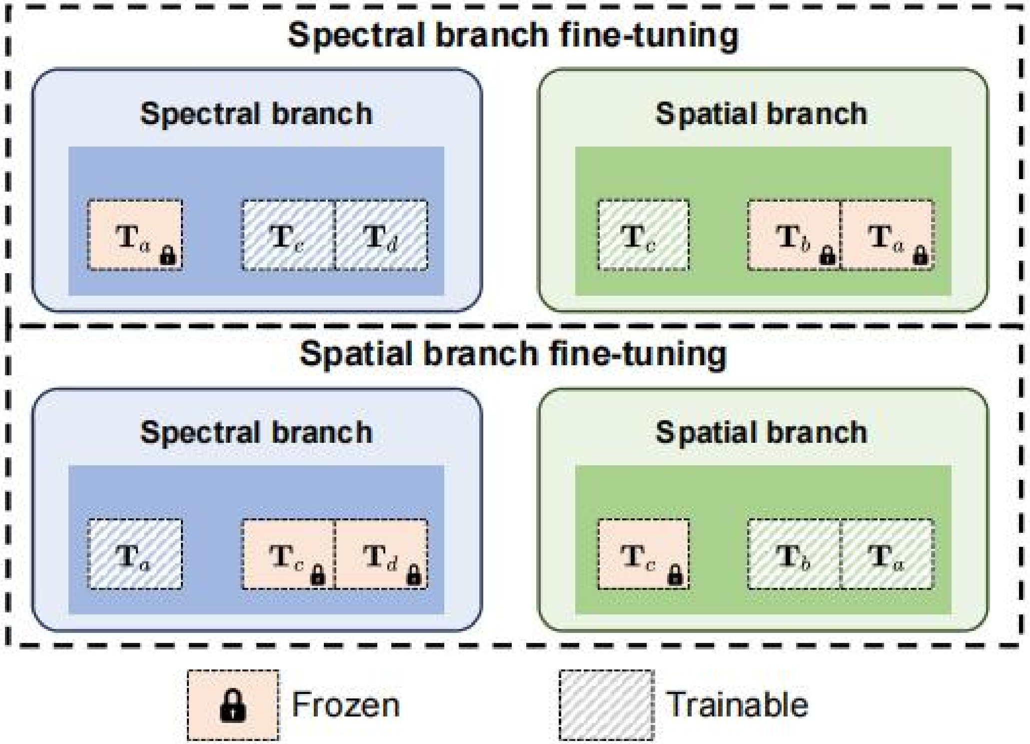


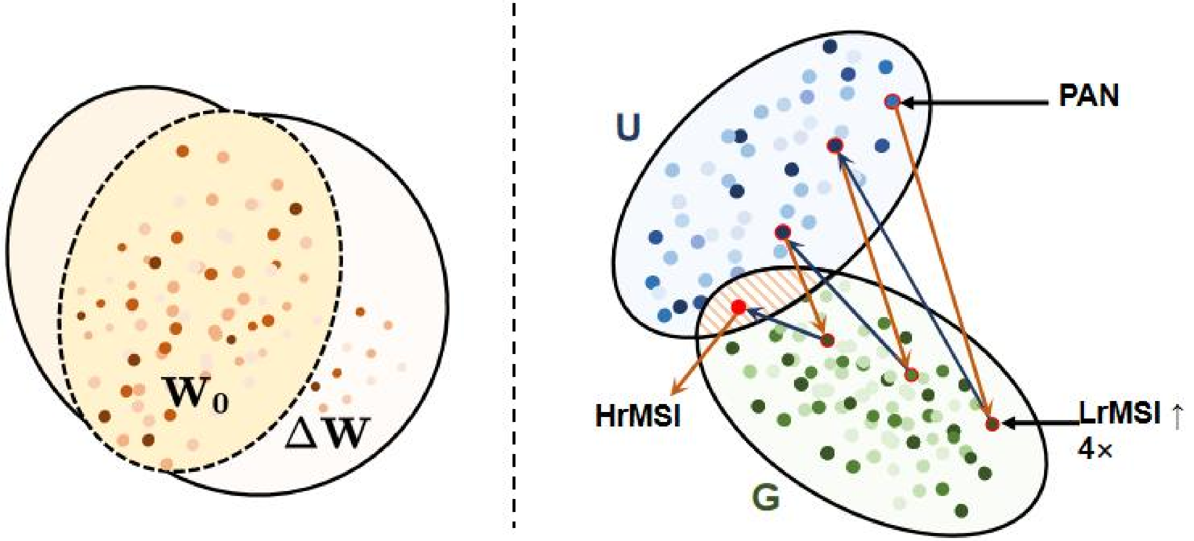
Theorem 1 *Assuming that $\mathcal{F}_{spa} \in \mathbb{R}^{H\times W\times S}$ and $\mathcal{F}_{spe} \in \mathbb{R}^{H\times W\times S}$ from the spatial and spectral branches, they can be alternatively projected as follows:*

$$\mathbf{T}^{spa} = Softmax\left(\frac{\mathbf{T}_a\mathbf{T}_b^T}{\sqrt{S'}}\right)\mathbf{T}_c^T,\ \ \mathbf{T}^{spe} = Softmax\left(\frac{\mathbf{T}_c\mathbf{T}_d^T}{\sqrt{\frac{(S')^3}{HW}}}\right)\mathbf{T}_a^T, \quad (6)$$

Methodology

LoRA-like Branch-wise Alternative Fine-tuning (L-BAF)





- The L-BAF method alternately fine-tunes the spatial and spectral branches **based on APFM**.

Training Stage of SSDiff Model:

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

Algorithm 1: Training stage of the proposed method.

Data: GT image $\mathbf{x}_0$, diffusion model $\mathbf{x}_\theta$ with its parameters θ, spectral and spatial branch parameter θ_{spe}, θ_{spa}, respectively, condition $cond$, timestep t, and denoised objective $\hat{\mathbf{x}}_0$.

Result: Optimized diffusion model $\mathbf{x}_\theta^*$.

$\mathrm{cond} \leftarrow \mathrm{PAN}, \mathrm{LrMSI}, \mathbf{x_t}$;

while *until convergence* **do**

  $t \leftarrow \mathrm{Uniform}(0, T)$; $\epsilon \sim \mathcal{N}(0, \boldsymbol{I})$;

  $\mathbf{x}_t \leftarrow \sqrt{\bar{\alpha}_t}(\mathbf{x}_0 - \mathrm{LrMSI}) + \sqrt{1 - \bar{\alpha}_t}\epsilon$; $\hat{\mathbf{x}}_0 \leftarrow \mathbf{x}_\theta(\mathbf{x}_t, \mathbf{cond}) + \mathrm{LrMSI}$;

  if *iteration* $>$ *150k* **then**

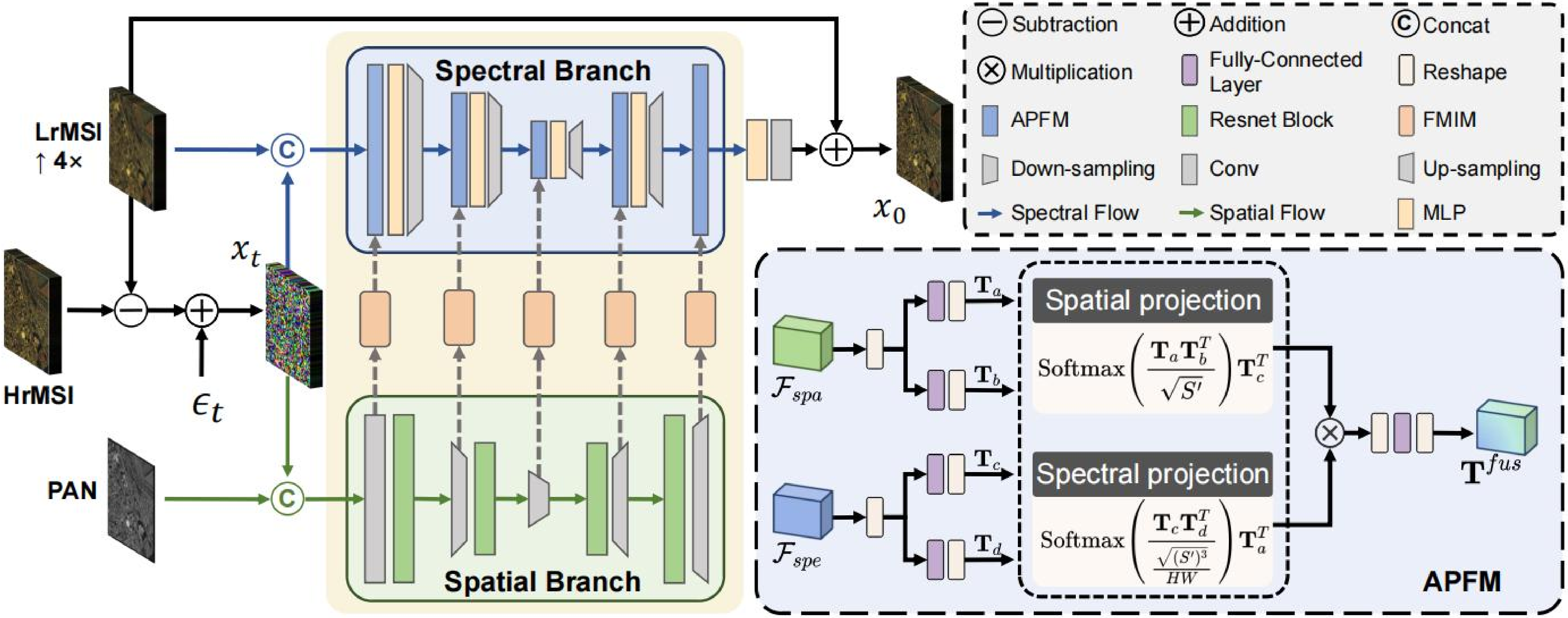
    fine-tune θ_{spe} or θ_{spa}; `// L-BAF`

  end

  $\theta \leftarrow \nabla_\theta \mathcal{L}_{simple}(\hat{\mathbf{x}}_0, \mathbf{x}_0)$.

end

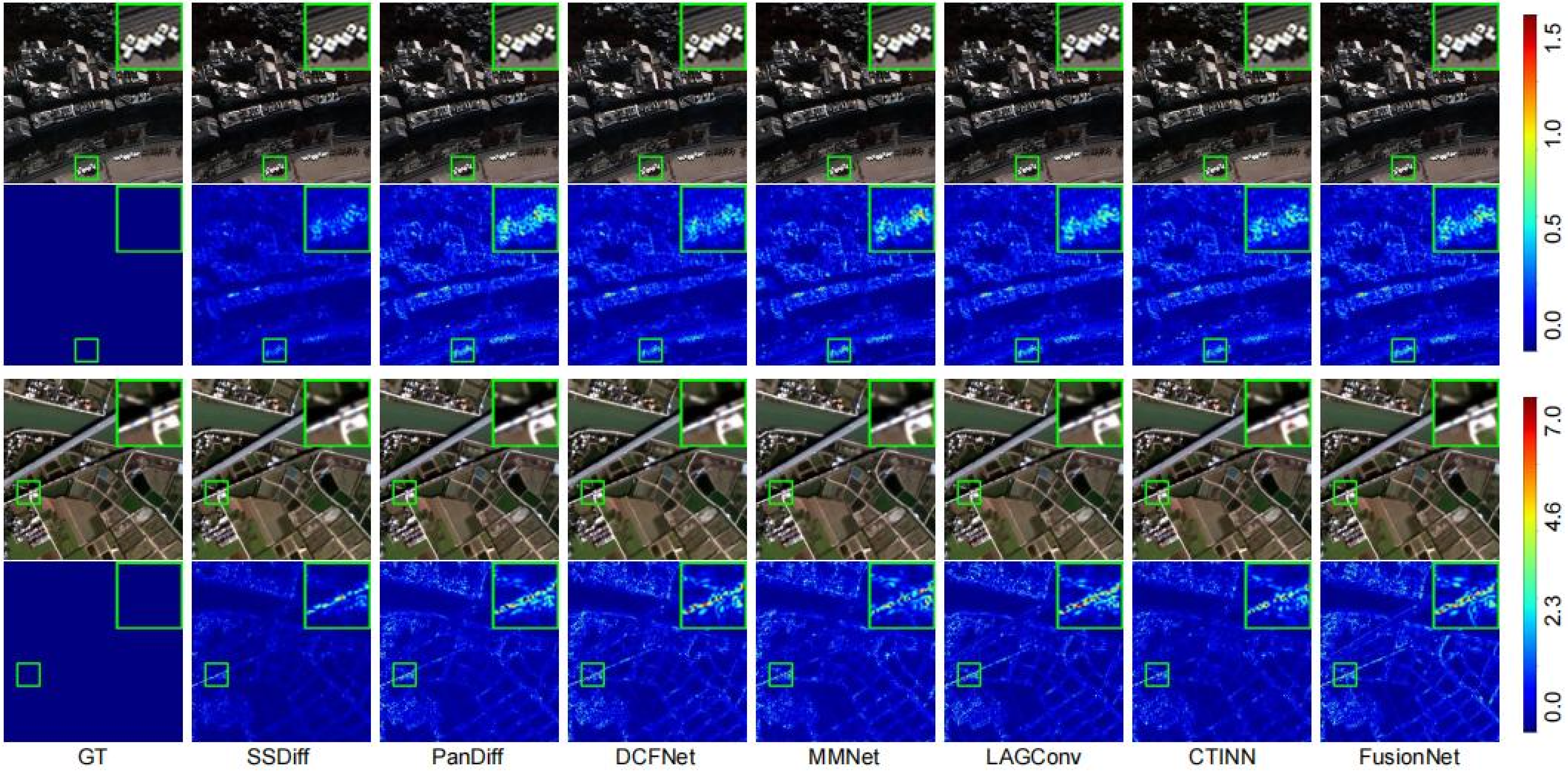
The detailed flowchart of our proposed method:



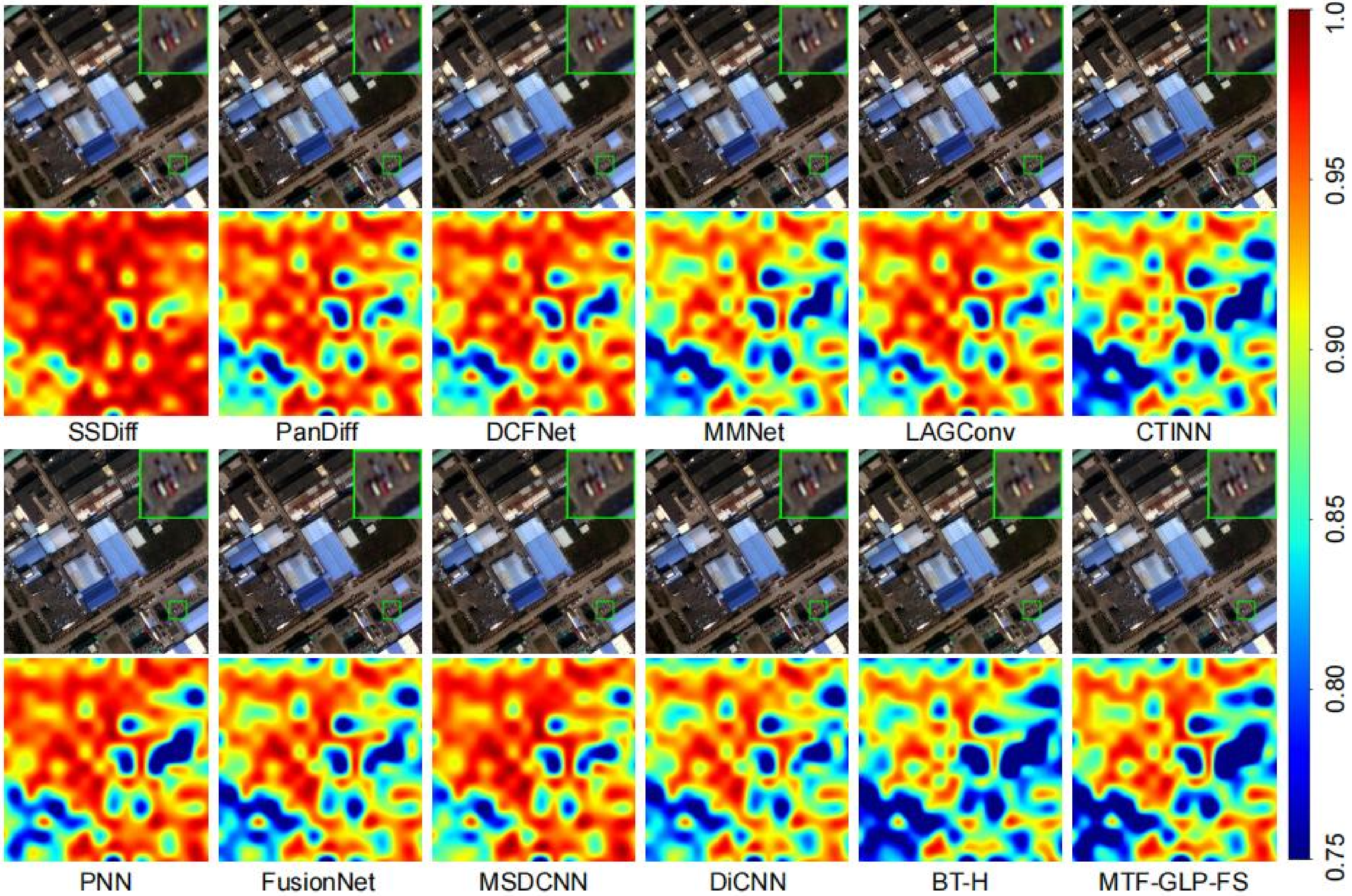
Quantitative comparison: WorldView-3 and GaoFen-2

Method	Reduced resolution				Full resolution		
	SAM(± std)	ERGAS(± std)	$Q2^n$(± std)	SCC(± std)	D_λ(± std)	D_s(± std)	HQNR(± std)
BDSD-PC [31]	5.4675±1.7185	4.6549±1.4667	0.8117±0.1063	0.9049±0.0419	0.0625±0.0235	0.0730±0.0356	0.8698±0.0531
MTF-GLP-FS [33]	5.3233±1.6548	4.6452±1.4441	0.8177±0.1014	0.8984±0.0466	0.0206±0.0082	0.0630±0.0284	0.9180±0.0346
BT-H [1]	4.8985±1.3028	4.5150±1.3315	0.8182±0.1019	0.9240±0.0243	0.0574±0.0232	0.0810±0.0374	0.8670±0.0540
PNN [20]	3.6798±0.7625	2.6819±0.6475	0.8929±0.0923	0.9761±0.0075	0.0213±0.0080	0.0428±0.0147	0.9369±0.0212
DiCNN [12]	3.5929±0.7623	2.6733±0.6627	0.9004±0.0871	0.9763±0.0072	0.0362±0.0111	0.0462±0.0175	0.9195±0.0258
MSDCNN [36]	3.7773±0.8032	2.7608±0.6884	0.8900±0.0900	0.9741±0.0076	0.0230±0.0091	0.0467±0.0199	0.9316±0.0271
FusionNet [5]	3.3252±0.6978	2.4666±0.6446	0.9044±0.0904	0.9807±0.0069	0.0239±0.0090	0.0364±0.0137	0.9406±0.0197
CTINN [48]	3.2523±0.6436	2.3936±0.5194	0.9056±0.0840	0.9826±0.0046	0.0550±0.0288	0.0679±0.0312	0.8815±0.0488
LAGConv [15]	3.1042±0.5585	2.2999±0.6128	0.9098±0.0907	0.9838±0.0068	0.0368±0.0148	0.0418±0.0152	0.9230±0.0247
MMNet [49]	3.0844±0.6398	2.3428±0.6260	0.9155±0.0855	0.9829±0.0056	0.0540±0.0232	0.0336±0.0115	0.9143±0.0281
DCFNet [39]	3.0264±0.7397	2.1588±0.4563	0.9051±0.0881	0.9861±0.0038	0.0781±0.0812	0.0508±0.0342	0.8771±0.1005
PanDiff [21]	3.2968±0.6010	2.4667±0.5837	0.8980±0.0880	0.9800±0.0063	0.0273±0.0123	0.0542±0.0264	0.9203±0.0360
SSDiff (ours)	**2.8429±0.5284**	**2.1059±0.4560**	**0.9156±0.0841**	**0.9867±0.0038**	**0.0132±0.0049**	**0.0307±0.0029**	**0.9565±0.0057**
BDSD-PC [31]	1.7110±0.3210	1.7025±0.4056	0.9932±0.0308	0.9448±0.0166	0.0759±0.0301	0.1548±0.0280	0.7812±0.0409
MTF-GLP-FS [33]	1.6757±0.3457	1.6023±0.3545	0.8914±0.0256	0.9390±0.0197	0.0336±0.0129	0.1404±0.0277	0.8309±0.0334
BT-H [1]	1.6810±0.3168	1.5524±0.3642	0.9089±0.0292	0.9508±0.0150	0.0602±0.0252	0.1313±0.0193	0.8165±0.0305
PNN [20]	1.0477±0.2264	1.0572±0.2355	0.9604±0.0100	0.9772±0.0054	0.0367±0.0291	0.0943±0.0224	0.8726±0.0373
DiCNN [12]	1.0525±0.2310	1.0812±0.2510	0.9594±0.0101	0.9771±0.0058	0.0413±0.0128	0.0992±0.0131	0.8636±0.0165
MSDCNN [36]	1.0472±0.2210	1.0413±0.2309	0.9612±0.0108	0.9782±0.0050	0.0269±0.0131	0.0730±0.0093	0.9020±0.0128
FusionNet [5]	0.9735±0.2117	0.9878±0.2222	0.9641±0.0093	0.9806±0.0049	0.0400±0.0126	0.1013±0.0134	0.8628±0.0184
CTINN [48]	0.8251±0.1386	0.6995±0.1068	0.9772±0.0117	0.9803±0.0015	0.0586±0.0260	0.1096±0.0149	0.8381±0.0237
LAGConv [15]	0.7859±0.1478	0.6869±0.1125	0.9804±0.0085	0.9906±0.0019	0.0324±0.0130	0.0792±0.0136	0.8910±0.0204
MMNet [49]	0.9929±0.1411	0.8117±0.1185	0.9690±0.0204	0.9859±0.0024	0.0428±0.0300	0.1033±0.0129	0.8583±0.0269
DCFNet [39]	0.8896±0.1577	0.8061±0.1369	0.9727±0.0100	0.9853±0.0024	0.0234±0.0116	0.0659±0.0096	0.9122±0.0119
PanDiff [21]	0.8881±0.1197	0.7461±0.1032	0.9792±0.0097	0.9887±0.0020	0.0265±0.0195	0.0729±0.0103	0.9025±0.0209
SSDiff (ours)	**0.6694±0.1244**	**0.6038±0.1080**	**0.9836±0.0074**	**0.9915±0.0017**	**0.0164±0.0093**	**0.0267±0.0071**	**0.9573±0.0100**
Ideal value	**0**	**0**	**1**	**1**	**0**	**0**	**1**

Qualitative comparison: WorldView-3 and GaoFen-2 reduced resolution



Qualitative comparison: GaoFen-2 full resolution



Thanks for your attention !

School of Mathematical Sciences,
University of Electronic Science and Technology of China (UESTC)

Code: **https://github.com/Z-ypnos/SSdiff_main**
Datasets: **https://github.com/liangjiandeng/PanCollection**