Regulating Rather than Constraining: Adaptive Guidance for Complex Spectral Reconstruction in Pansharpening

Supplementary Material

Abstract

This supplementary material provides additional details and extended analyses to support the findings presented in the main text. We first elaborate on the specifics of the datasets employed and the implementation details of our training procedure. To comprehensively validate the efficacy of our method, we then present extensive comparative studies against state-of-the-art approaches, alongside a series of ablation experiments that dissect the contribution of each key component (Tables 1-10). Furthermore, complementary visualizations (Figures 1-8) are provided to offer intuitive insights into the model's performance and behavior.

A. Datasets

We conduct experiments on three satellite datasets: WorldView-3 (WV3), QuickBird (QB), and Gaofen-2 (GF2). Each dataset consists of image patches cropped from large remote sensing scenes and is split into training and testing subsets. The training set contains triplets of simulated panchromatic images (PAN), low-resolution multispectral images (LRMS), and the corresponding ground truth (GT), with spatial sizes of $64 \times 64 \times 1$, $16 \times 16 \times C$, and $64 \times 64 \times C$, respectively. The WV3 training set includes approximately 10,000 samples with 8 spectral channels ($C = 8$); the QB training set contains around 17,000 samples with 4 channels ($C = 4$); and the GF2 dataset provides 20,000 samples, also with 4 channels ($C = 4$).

For each satellite, the reduced-resolution test set includes 20 PAN/LRMS/GT triplets covering diverse land-cover types. These downsampled test samples have sizes of $256 \times 256 \times 1$, $64 \times 64 \times C$, and $256 \times 256 \times C$. In the full-resolution testing stage, each dataset offers 20 PAN/LRMS pairs with sizes of $512 \times 512 \times 1$ and $128 \times 128 \times C$, respectively. All datasets and their preprocessing pipelines follow the PanCollection repository [2].

B. Traning Details

In the comparative experiments, we employed the Adam optimizer [7] with a fixed learning rate of 0.0003. The parameters of the Beta distribution in MixShuffle (θ_1, θ_2) were set to $(0.4, 0.2)$, and the hyperparameter γ in HAL was set to 2. For specific datasets: on WV3, the batch size was set to 32 with 270,000 training iterations, the MixShuffle probability intensity (p_1, p_2) was $(0.8, 0.8)$, and the HAL weight coefficients $(\lambda_{\text{pixel}}, \lambda_{\text{channel}}, \lambda_{\text{sample}})$ were $(0.5, 0.25, 0.25)$; on GF2, the batch size was set to 48 with 795,000 training iterations, the MixShuffle probability intensity remained $(0.8, 0.8)$, while the HAL weight coefficients were adjusted to $(0.6, 0.2, 0.2)$; on QB, the batch size was set to 48 with 85,000 training iterations, the MixShuffle probability intensity was set to $(0.8, 0.5)$, and the HAL weight coefficients were $(0.5, 0.3, 0.2)$. Random rotation data augmentation with a probability of 0.4 was also applied to the WV3 and GF2 datasets. More detailed parameter settings can be found in the open-source code.

To ensure a fair comparison, all deep learning models were trained and evaluated on the same datasets using the default configurations recommended in their respective publications. When integrating our proposed regularization strategy into different baseline models, we maintained the original experimental setups as a priority while keeping the parameter settings of the regularization strategy consistent with those used in DANet. It is worth noting that only a few baseline models required minor parameter adjustments, whereas most models directly applied our regularization strategy and achieved significant performance improvements.

C. Additional Results

C.1. Quantitative Results

Table 1 and Table 2 present the performance comparison of various baseline methods before and after introducing our proposed regularization strategy on the WV3 and GF2 datasets, respectively. These baselines cover different architectures including CNN, Transformer, and Mamba. The results demonstrate that incorporating our regularization strategy leads to significant performance improvements across all baseline models on different datasets, highlighting its strong compatibility with diverse architectures and data distributions. For instance, on the GF2 dataset, our regularization method improved the performance of all baselines by 4.41%–13.48% and 5.00%–16.00% on the two key metrics, SAM and ERGAS, respectively.

Table 3 provides a comprehensive quantitative analysis focusing on typical spectral mixing regions – ground object boundaries. It shows that our method achieves substantial gains in these challenging areas, regardless of the backbone architecture or dataset. For example, on the WV3, GF2, and QB datasets, it enhanced the boundary accuracy of FusionNet [1] by 18.74%, 19.65%, and 19.03%, respectively. This observation aligns well with the motivation behind our tailored regularization techniques, MixShuffle and HAL, de-

Table 1. Comparison of baselines on the WV3 dataset. Methods marked with * are integrated with our approach. Best results are in bold.

	Reduced Resolution (RR): Avg±std				Full Resolution (FR): Avg±std			
	SAM↓	ERGAS↓	Q2n↑	SCC↑	D_λ↓	D_s↓	QNR↑	HQNR↑
FusionNet [1]	3.31±0.63	2.45±0.59	0.896±0.093	0.980±0.006	0.024±0.009	0.036±0.014	0.941±0.018	0.941±0.020
FusionNet*	**2.92±0.57**	**2.18±0.48**	**0.912±0.083**	**0.984±0.008**	**0.021±0.007**	**0.034±0.016**	**0.946±0.022**	**0.945±0.026**
LAGNet [6]	3.10±0.50	2.29±0.56	0.902±0.091	0.983±0.006	0.037±0.015	0.042±0.015	0.922±0.027	0.923±0.025
LAGNet*	**2.89±0.46**	**2.16±0.58**	**0.909±0.086**	**0.985±0.005**	**0.029±0.019**	**0.038±0.017**	**0.934±0.024**	**0.934±0.030**
Panformer [15]	3.27±0.69	2.37±0.54	0.906±0.082	0.984±0.009	0.045±0.036	0.067±0.012	0.891±0.035	0.892±0.039
Panformer*	**2.97±0.73**	**2.18±0.47**	**0.913±0.077**	**0.986±0.010**	**0.039±0.024**	**0.059±0.016**	**0.904±0.041**	**0.905±0.038**
Invformer [16]	3.25±0.64	2.39±0.52	0.906±0.084	0.983±0.005	0.055±0.029	0.068±0.031	0.881±0.053	0.882±0.049
Invformer*	**3.02±0.58**	**2.23±0.56**	**0.914±0.098**	**0.985±0.009**	**0.049±0.031**	**0.065±0.034**	**0.889±0.056**	**0.890±0.047**
PanMamba [4]	2.94±0.54	2.20±0.51	0.916±0.090	0.985±0.006	**0.020±0.007**	0.031±0.003	0.950±0.065	0.951±0.070
PanMamba*	**2.85±0.55**	**2.09±0.52**	**0.919±0.059**	**0.988±0.008**	0.026±0.011	**0.024±0.008**	**0.951±0.056**	**0.952±0.072**
FusionMamba [8]	2.82±0.64	2.11±0.49	**0.920±0.091**	**0.989±0.005**	0.019±0.008	0.027±0.006	0.955±0.019	0.955±0.016
FusionMamba*	**2.80±0.57**	**1.95±0.38**	0.919±0.085	**0.989±0.006**	**0.018±0.007**	**0.025±0.009**	**0.957±0.023**	**0.956±0.019**
ADWM [11]	2.89±0.91	1.98±0.72	0.916±0.140	**0.989±0.006**	**0.024±0.020**	0.033±0.028	0.944±0.043	0.945±0.019
ADWM*	**2.81±0.82**	**1.94±0.68**	**0.918±0.122**	0.988±0.009	0.026±0.018	**0.027±0.026**	**0.948±0.056**	**0.949±0.023**

Table 2. Comparison of baselines on the GF2 dataset. Methods marked with * are integrated with our approach. Best results are in bold.

	Reduced Resolution (RR): Avg±std				Full Resolution (FR): Avg±std			
	SAM↓	ERGAS↓	Q2n↑	SCC↑	D_λ↓	D_s↓	QNR↑	HQNR↑
FusionNet [1]	0.98±0.19	1.00±0.20	0.978±0.005	0.978±0.005	0.040±0.013	0.101±0.013	0.863±0.023	0.863±0.018
FusionNet*	**0.85±0.17**	**0.84±0.28**	**0.980±0.007**	**0.981±0.006**	**0.036±0.015**	**0.089±0.014**	**0.878±0.021**	**0.879±0.017**
LAGNet [6]	0.80±0.14	0.71±0.11	0.979±0.011	0.989±0.002	0.032±0.013	0.079±0.014	0.891±0.032	0.891±0.020
LAGNet*	**0.75±0.13**	**0.63±0.13**	**0.981±0.009**	**0.991±0.005**	**0.028±0.014**	**0.065±0.019**	**0.909±0.029**	**0.908±0.023**
Panformer [15]	0.89±0.19	0.69±0.14	0.976±0.012	0.982±0.005	0.065±0.025	0.121±0.019	0.822±0.025	0.821±0.032
Panformer*	**0.77±0.21**	**0.58±0.15**	**0.978±0.014**	**0.993±0.007**	**0.057±0.021**	**0.098±0.023**	**0.851±0.017**	**0.850±0.028**
Invformer [16]	0.83±0.14	0.70±0.11	0.977±0.012	0.980±0.002	0.059±0.026	0.110±0.015	0.837±0.022	0.838±0.024
Invformer*	**0.73±0.12**	**0.62±0.16**	**0.978±0.013**	**0.985±0.007**	**0.054±0.028**	**0.093±0.017**	**0.858±0.023**	**0.859±0.029**
PanMamba [4]	0.68±0.12	0.64±0.10	0.982±0.008	**0.985±0.006**	**0.016±0.008**	0.045±0.009	0.940±0.016	0.939±0.010
PanMamba*	**0.65±0.13**	**0.60±0.12**	**0.984±0.010**	**0.985±0.005**	0.021±0.009	**0.036±0.011**	**0.944±0.014**	**0.944±0.012**
FusionMamba [8]	0.71±0.14	0.62±0.11	0.984±0.007	0.995±0.009	**0.017±0.009**	0.030±0.007	0.952±0.008	0.952±0.010
FusionMamba*	**0.67±0.12**	**0.56±0.13**	**0.985±0.005**	**0.996±0.012**	0.019±0.008	**0.026±0.010**	**0.955±0.009**	**0.956±0.013**
ADWM [11]	0.68±0.18	0.60±0.16	0.984±0.016	0.996±0.002	0.033±0.029	0.050±0.033	0.920±0.055	0.921±0.047
ADWM*	**0.64±0.14**	**0.57±0.18**	**0.985±0.014**	**0.988±0.018**	**0.031±0.027**	**0.044±0.037**	**0.926±0.046**	**0.925±0.052**

signed specifically for pansharpening. Table 4 shows the comparative experimental results on the QB dataset. Regarding the definition of Boundary ERGAS, we use the Sobel edge detector applied to the reference image to extract boundary regions. Specifically, pixels with the top 20% of Sobel gradient magnitudes are selected as boundaries, and the Boundary ERGAS is calculated only on these areas.

Table 5 details the complete ablation studies, evaluating the individual and combined contributions of MixShuffle and HAL. The results indicate that both components independently contribute to performance improvement. Specifically, introducing either MixShuffle or HAL alone leads to consistent improvements across all evaluation metrics on all three datasets. More importantly, their joint utilization yields the best performance. For instance, on the WV3 dataset, our full model equipped with both components significantly reduced the ERGAS score from 2.07 to 1.91 compared to the baseline. In addition, Table 6 presents the results of the component-level ablation study on MixShuffle, demonstrating the necessity of sample-level mixing and channel-level shuffling within MixShuffle.

Tables 7-10 present the comparison results for different hyperparameter configurations in MixShuffle and HAL, respectively. As shown in Table 7, the intensity of the random convex combination influences network performance in a patterned manner: neither excessively high nor low intensity can fully unleash the network's learning potential, with the optimal performance achieved when the random intensity parameters (p_1, p_2) are set to $(0.8, 0.5)$. Regarding the β-distribution parameters, the results in Table 8 demonstrate that MixShuffle remains stable with respect to θ_1 and θ_2, requiring no careful tuning. As for HAL, Table 9 shows that it achieves the best performance when $\gamma = 2$, with only slight degradation observed for other values. Finally, Table 10 illustrates the impact of different loss weights, specifically at the pixel, channel, and sample levels, on the final

Table 3. ERGAS results on WV3, GF2, and QB datasets. Asterisked (*) methods are our enhanced baselines; best scores are bolded.

	WV3: Avg±std		GF2: Avg±std		QB: Avg±std	
	Overall	**Boundary**	**Overall**	**Boundary**	**Overall**	**Boundary**
FusionNet [1]	2.45±0.59	10.78±4.12	1.00±0.20	3.46±0.94	4.19±0.25	14.03±3.23
FusionNet*	**2.18±0.48**	**8.76±3.94**	**0.84±0.28**	**2.78±0.48**	**4.06±0.22**	**11.36±2.76**
Improvement (%)	11.02	18.74	16.00	19.65	3.10	19.03
LAGNet [6]	2.29±0.56	9.28±4.18	0.71±0.11	2.95±0.87	3.87±0.36	12.61±2.89
LAGNet*	**2.16±0.58**	**8.79±3.97**	**0.63±0.13**	**2.56±0.79**	**3.69±0.42**	**11.82±3.01**
Improvement (%)	5.68	5.28	11.27	13.22	4.65	6.26
PanMamba [4]	2.20±0.51	9.16±4.23	0.64±0.10	2.82±0.67	4.38±0.60	14.95±3.57
PanMamba*	**2.12±0.54**	**8.77±3.86**	**0.60±0.12**	**2.69±0.58**	**4.28±0.71**	**14.17±3.15**
Improvement (%)	3.64	4.26	6.25	4.61	2.28	5.22
FusionMamba [8]	2.09±0.51	8.79±4.03	0.62±0.11	2.69±0.64	4.61±0.47	15.69±3.42
FusionMamba*	**1.95±0.41**	**7.46±3.75**	**0.56±0.13**	**2.44±0.67**	**4.53±0.51**	**15.03±3.43**
Improvement (%)	6.70	15.13	9.68	9.29	1.74	4.21
Panformer [15]	2.37±0.54	9.48±4.22	0.69±0.14	2.89±0.77	3.82±0.25	12.78±2.93
Panformer*	**2.18±0.47**	**8.74±3.89**	**0.58±0.15**	**2.51±0.68**	**3.68±0.31**	**12.04±2.75**
Improvement (%)	8.02	7.81	15.94	13.15	3.66	5.79
Invformer [16]	2.39±0.52	9.94±4.13	0.70±0.11	2.91±0.69	3.70±0.29	12.36±2.75
Invformer*	**2.23±0.56**	**9.22±3.96**	**0.62±0.16**	**2.47±0.65**	**3.59±0.30**	**11.32±2.89**
Improvement (%)	6.69	7.24	11.43	15.12	2.97	8.41

Table 4. Quantitative comparison of different pansharpening methods on the QB dataset. The best-performing results are in bold.

	Reduced Resolution (RR): Avg±std				Full Resolution (FR): Avg±std			
	SAM↓	**ERGAS↓**	**Q2n↑**	**SCC↑**	D_λ↓	D_s↓	**QNR↑**	**HQNR↑**
MTF-GLP-FS [10]	7.87±1.67	7.45±0.56	0.834±0.096	0.902±0.025	0.049±0.015	0.138±0.024	0.820±0.042	0.820±0.034
LRTCFPan [13]	7.29±1.60	7.03±0.62	0.853±0.095	0.914±0.014	**0.023±0.012**	0.071±0.035	0.908±0.051	0.909±0.044
PanNet [14]	5.88±1.07	6.02±0.79	0.881±0.102	0.947±0.018	0.041±0.011	0.114±0.032	0.850±0.038	0.850±0.039
FusionNet [1]	4.96±0.84	4.19±0.25	0.923±0.097	0.976±0.010	0.059±0.019	0.052±0.009	0.892±0.031	0.892±0.022
LAGNet [6]	4.58±0.77	3.87±0.36	0.932±0.094	0.981±0.009	0.084±0.024	0.068±0.014	0.854±0.021	0.854±0.018
Invformer [16]	4.66±0.78	3.70±0.29	0.932±0.007	0.983±0.007	0.174±0.033	0.073±0.024	0.766±0.038	0.766±0.043
DCFNet [12]	4.54±0.73	3.85±0.28	0.932±0.093	0.974±0.010	0.045±0.015	0.124±0.027	0.837±0.020	0.836±0.016
HMPNet [9]	4.72±0.38	3.66±0.40	0.930±0.110	**0.988±0.009**	0.183±0.054	0.079±0.025	0.752±0.062	0.754±0.065
CANNet [3]	4.54±0.79	3.74±0.31	**0.935±0.089**	0.982±0.007	0.038±0.013	0.047±0.009	0.916±0.020	0.917±0.012
PanMamba [4]	4.74±0.88	4.38±0.60	0.923±0.092	0.975±0.011	0.049±0.013	0.044±0.016	0.909±0.034	0.910±0.027
FusionMamba [8]	4.72±0.96	4.61±0.47	0.925±0.131	0.973±0.012	0.040±0.035	0.052±0.042	0.912±0.063	0.911±0.049
ADWM [11]	4.48±0.87	3.67±0.59	**0.935±0.109**	0.983±0.010	0.038±0.036	0.045±0.041	0.919±0.062	0.918±0.054
ARNet [5]	4.43±0.81	**3.63±0.33**	0.934±0.092	0.983±0.012	0.038±0.015	**0.040±0.009**	**0.923±0.024**	**0.924±0.019**
Proposed	**4.38±0.64**	3.67±0.43	**0.935±0.076**	0.984±0.013	0.036±0.020	0.053±0.021	0.914±0.023	0.915±0.018

performance of HAL.

C.2. Visualization

Fig. 1 and Fig. 2 reveal the reconstruction error disparities among different methods through comparisons across various regions and spectral channels. The limitations of existing methods are most pronounced at the object boundaries and textural areas, which is attributed to the inherent reconstruction difficulty caused by complex spectral confusion. In contrast, our method achieves the lowest error levels across all comparative scenarios and spectral channels. Particularly in challenging regions with severe spectral mixing, such as between building roofs and the ground, our approach effectively overcomes the reconstruction challenges by accurately modeling the intrinsic spectral mixing mechanisms.

As shown in the visual assessment in Fig. 3, integrating our method consistently enhances the reconstruction quality across various baseline architectures. This improvement is particularly pronounced at various land-cover boundaries, such as the edges of building roofs and the junctions between different materials within the same structure. Due to the diversity of materials and design styles, these areas exhibit complex spectral mixing patterns. Our regularization strategy, by accurately modeling these patterns, successfully drives consistent performance improvements in such challenging regions across different architectures.

As shown in Fig. 4, Fig. 5, Fig. 6, and Fig. 7, the gradient contributions of the proposed HAL and the L1 loss are compared throughout the training process. The analysis reveals that HAL achieves a dynamic optimization mechanism: during the early stages of training, its gradients are highly concentrated in key regions with significant spectral mixing (such as object boundaries), thereby compelling the network to enhance feature learning in these areas. Conversely, in the later stages of training, the gradient contribu-

Table 5. Ablation experiment on WV3, GF2, and QB datasets. The best values are highlighted in bold.

	MixShuffle	HAL	Reduced Resolution (RR): Avg±std				Full Resolution (FR): Avg±std			
			SAM↓	ERGAS↓	Q2n↑	SCC↑	D_λ↓	D_s↓	QNR↑	HQNR↑
WV3			2.85±0.52	2.07±0.42	0.912±0.110	0.988±0.012	0.027±0.012	0.031±0.008	0.943±0.022	0.944±0.012
	✓		2.74±0.48	1.96±0.47	0.918±0.098	0.990±0.011	0.024±0.013	0.026±0.011	0.951±0.026	0.952±0.015
		✓	2.78±0.44	1.99±0.45	0.916±0.108	0.989±0.014	0.025±0.012	0.024±0.012	0.952±0.019	0.951±0.016
	✓	✓	**2.69±0.42**	**1.91±0.39**	**0.921±0.131**	**0.991±0.010**	**0.023±0.011**	**0.019±0.009**	**0.958±0.023**	**0.959±0.011**
GF2			0.62±0.12	0.59±0.09	0.983±0.013	0.994±0.005	0.031±0.011	0.041±0.009	0.929±0.021	0.929±0.015
	✓		0.59±0.14	0.55±0.11	0.986±0.011	0.997±0.006	0.027±0.012	0.033±0.010	0.941±0.022	0.942±0.016
		✓	0.58±0.12	0.57±0.13	0.985±0.010	**0.998±0.005**	0.028±0.013	0.036±0.011	0.937±0.023	0.938±0.014
	✓	✓	**0.58±0.13**	**0.53±0.10**	**0.987±0.009**	**0.998±0.002**	**0.024±0.010**	**0.024±0.007**	**0.953±0.018**	**0.953±0.013**
QB			4.60±0.55	4.10±0.54	0.926±0.081	0.979±0.008	0.044±0.023	0.070±0.022	0.889±0.021	0.888±0.021
	✓		4.43±0.67	3.70±0.45	0.934±0.077	0.982±0.011	0.038±0.025	0.053±0.021	0.911±0.019	0.912±0.021
		✓	4.45±0.59	3.72±0.47	0.934±0.072	0.981±0.009	0.040±0.019	0.054±0.026	0.908±0.024	0.908±0.023
	✓	✓	**4.38±0.64**	**3.67±0.43**	**0.935±0.076**	**0.984±0.013**	**0.036±0.020**	**0.053±0.021**	**0.914±0.023**	**0.915±0.018**

Table 6. Experimental results of the component-level ablation study on MixShuffle. The best values are highlighted in bold.

	Sample-level	Channel-level	Reduced Resolution (RR): Avg±std				Full Resolution (FR): Avg±std			
			SAM↓	ERGAS↓	Q2n↑	SCC↑	D_λ↓	D_s↓	QNR↑	HQNR↑
WV3			2.78±0.44	1.99±0.45	0.916±0.108	0.989±0.014	0.025±0.012	0.024±0.012	0.952±0.019	0.951±0.016
	✓		2.73±0.47	1.95±0.46	0.918±0.098	0.990±0.011	0.026±0.015	0.021±0.013	0.954±0.025	0.954±0.016
		✓	2.74±0.44	1.97±0.46	0.918±0.118	0.990±0.013	0.024±0.012	0.023±0.014	0.954±0.017	0.954±0.015
	✓	✓	**2.69±0.42**	**1.91±0.39**	**0.921±0.131**	**0.991±0.010**	**0.023±0.011**	**0.019±0.009**	**0.958±0.023**	**0.959±0.011**
GF2			0.62±0.12	0.59±0.09	0.983±0.013	0.994±0.005	0.031±0.011	0.041±0.009	0.929±0.021	0.929±0.015
	✓		0.59±0.14	0.55±0.11	0.986±0.011	0.997±0.006	0.027±0.012	0.033±0.010	0.941±0.022	0.942±0.016
		✓	0.58±0.12	0.57±0.13	0.985±0.010	**0.998±0.005**	0.028±0.013	0.036±0.011	0.937±0.023	0.938±0.014
	✓	✓	**0.58±0.13**	**0.53±0.10**	**0.987±0.009**	**0.998±0.002**	**0.024±0.010**	**0.024±0.007**	**0.953±0.018**	**0.953±0.013**
QB			4.45±0.59	3.72±0.47	0.934±0.072	0.981±0.009	0.040±0.019	0.054±0.026	0.908±0.024	0.908±0.023
	✓		4.42±0.64	3.70±0.45	0.934±0.074	0.982±0.012	0.038±0.025	0.053±0.020	0.911±0.019	0.912±0.022
		✓	4.43±0.61	3.69±0.47	0.934±0.071	0.982±0.010	0.037±0.019	0.055±0.025	0.910±0.024	0.910±0.023
	✓	✓	**4.38±0.64**	**3.67±0.43**	**0.935±0.076**	**0.984±0.013**	**0.036±0.020**	**0.053±0.021**	**0.914±0.023**	**0.915±0.018**

tion becomes more balanced, indicating that HAL successfully mitigates overfitting in challenging regions and consequently improves the model's generalization capability. In addition, Fig. 8 shows more examples of new training samples obtained through the MixShuffle transformation.

References

[1] 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*, page 6995–7010, 2021. 1, 2, 3

[2] Liang-jian Deng, Gemine Vivone, Mercedes E. Paoletti, Giuseppe Scarpa, Jiang He, Yongjun Zhang, Jocelyn Chanussot, and Antonio Plaza. Machine learning in pansharpening: A benchmark, from shallow to deep networks. *IEEE Geoscience and Remote Sensing Magazine*, 10(3): 279–315, 2022. 1

[3] Yule Duan, Xiao Wu, Haoyu Deng, and Liang-Jian Deng. Content-adaptive non-local convolution for remote sensing pansharpening. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 27738–27747, 2024. 3

[4] Xuanhua He, Ke Cao, Jie Zhang, Keyu Yan, Yingying Wang, Rui Li, Chengjun Xie, Danfeng Hong, and Man Zhou. Pan-mamba: Effective pan-sharpening with state space model. *Information Fusion*, 115:102779, 2025. 2, 3

[5] Jie Huang, Haorui Chen, Jiaxuan Ren, Siran Peng, and Liangjian Deng. A general adaptive dual-level weighting mechanism for remote sensing pansharpening. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 3

[6] 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. *Proceedings of the AAAI Conference on Artificial Intelligence*, page 1113–1121, 2022. 2, 3

[7] DiederikP. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv: Learning,arXiv: Learning*, 2014. 1

[8] Siran Peng, Xiangyu Zhu, Haoyu Deng, Liang-Jian Deng, and Zhen Lei. Fusionmamba: Efficient remote sensing image fusion with state space model. *IEEE Transactions on Geoscience and Remote Sensing*, page 1–1, 2024. 2, 3

[9] Xin Tian, Kun Li, Wei Zhang, Zhongyuan Wang, and Jiayi Ma. Interpretable model-driven deep network for hyperspectral, multispectral, and panchromatic image fusion. *IEEE Transactions on Neural Networks and Learning Systems*, 35 (10):14382–14395, 2024. 3

[10] Gemine Vivone, Rocco Restaino, and Jocelyn Chanussot. Full scale regression-based injection coefficients for

Table 7. Performance comparison of different hyperparameter configurations in MixShuffle on the QB Dataset. p_1 and p_2 denote the random convex combination probability intensities.

p_1	p_2	Reduced Resolution (RR): Avg±std				Full Resolution (FR): Avg±std			
		SAM↓	ERGAS↓	Q2n↑	SCC↑	D_λ↓	D_s↓	QNR↑	HQNR↑
0.8	0.8	4.52±0.53	4.00±0.47	0.928±0.076	0.979±0.008	0.042±0.024	0.061±0.023	0.900±0.022	0.899±0.021
0.8	0.7	4.46±0.58	3.88±0.48	0.932±0.072	0.980±0.009	0.040±0.020	0.056±0.026	0.906±0.024	0.906±0.023
0.8	0.6	4.44±0.68	3.76±0.46	0.933±0.071	0.981±0.010	0.041±0.019	0.054±0.026	0.907±0.024	0.907±0.023
0.8	0.5	**4.43±0.67**	**3.70±0.45**	**0.934±0.077**	**0.982±0.011**	**0.038±0.025**	**0.053±0.021**	**0.911±0.019**	**0.912±0.021**
0.8	0.4	4.45±0.65	3.75±0.45	0.933±0.062	0.980±0.011	0.042±0.017	0.055±0.025	0.905±0.022	0.906±0.023
0.8	0.3	4.47±0.59	3.90±0.55	0.931±0.057	0.979±0.007	0.043±0.019	0.059±0.024	0.901±0.025	0.902±0.019
0.8	0.2	4.51±0.56	3.96±0.49	0.927±0.061	0.978±0.011	0.043±0.015	0.059±0.024	0.901±0.022	0.901±0.024

Table 8. Performance comparison of different hyperparameter configurations in MixShuffle on the QB Dataset. θ_1 and θ_2 are β-distribution parameters.

θ_1	θ_2	Reduced Resolution (RR): Avg±std				Full Resolution (FR): Avg±std			
		SAM↓	ERGAS↓	Q2n↑	SCC↑	D_λ↓	D_s↓	QNR↑	HQNR↑
0.3	0.2	4.42±0.51	3.74±0.47	0.933±0.073	0.983±0.008	0.038±0.023	0.056±0.023	0.908±0.023	0.908±0.022
0.3	0.3	4.41±0.57	3.75±0.48	0.933±0.073	0.983±0.009	0.038±0.020	0.057±0.025	0.907±0.024	0.908±0.023
0.4	0.2	4.40±0.62	3.74±0.44	0.934±0.061	0.984±0.011	0.037±0.019	0.057±0.025	0.908±0.023	0.909±0.021
0.4	0.3	4.40±0.67	3.74±0.45	0.932±0.075	0.983±0.011	0.035±0.024	0.061±0.023	0.906±0.020	0.907±0.022
0.5	0.2	4.41±0.62	3.75±0.45	0.933±0.062	0.983±0.013	0.038±0.016	0.057±0.025	0.907±0.022	0.907±0.023
0.5	0.3	4.42±0.53	3.75±0.55	0.934±0.054	0.984±0.008	0.038±0.018	0.057±0.024	0.907±0.025	0.907±0.019

panchromatic sharpening. *IEEE Transactions on Image Processing*, page 3418–3431, 2018. 3

[11] Xueyang Wang, Zhixin Zheng, Jiandong Shao, Yule Duan, and Liang-Jian Deng. Adaptive rectangular convolution for remote sensing pansharpening. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17872–17881, 2025. 2, 3

[12] Xiao Wu, Ting-Zhu Huang, Liang-Jian Deng, and Tian-Jing Zhang. Dynamic cross feature fusion for remote sensing pansharpening. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 3

[13] 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. 3

[14] Junfeng Yang, Xueyang Fu, Yuwen Hu, Yue Huang, Xinghao Ding, and John Paisley. Pannet: A deep network architecture for pan-sharpening. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1753–1761, 2017. 3

[15] Huanyu Zhou, Qingjie Liu, and Yunhong Wang. Panformer: A transformer based model for pan-sharpening. In *2022 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2022. 2, 3

[16] Man Zhou, Xueyang Fu, Jie Huang, Feng Zhao, Aiping Liu, and Rujing Wang. Effective pan-sharpening with transformer and invertible neural network. *IEEE Transactions on Geoscience and Remote Sensing*, page 1–15, 2022. 2, 3

Table 9. Performance comparison of different hyperparameter configurations in HAL on the QB Dataset. γ is the parameter of the weighting function.

γ	Reduced Resolution (RR): Avg±std				Full Resolution (FR): Avg±std			
	SAM↓	ERGAS↓	Q2n↑	SCC↑	D_λ↓	D_s↓	QNR↑	HQNR↑
0	4.45±0.67	3.83±0.45	0.929±0.077	0.981±0.011	0.040±0.025	0.062±0.021	0.900±0.019	0.901±0.021
1	4.43±0.55	3.76±0.49	0.931±0.067	0.981±0.011	0.038±0.021	0.059±0.025	0.905±0.022	0.905±0.023
2	**4.39±0.63**	**3.73±0.42**	**0.935±0.060**	**0.984±0.013**	**0.037±0.020**	**0.056±0.024**	**0.909±0.022**	**0.909±0.024**
3	4.51±0.65	3.85±0.46	0.930±0.074	0.981±0.012	0.038±0.024	0.059±0.026	0.905±0.021	0.905±0.022

Table 10. Performance comparison of different hyperparameter configurations in HAL on the QB Dataset. λ_{pixel}, λ_{channel}, λ_{sample} represent the weights for pixel-level, channel-level, and sample-level losses, respectively.

(λ_{pixel}, λ_{channel}, λ_{sample})	Reduced Resolution (RR): Avg±std				Full Resolution (FR): Avg±std			
	SAM↓	ERGAS↓	Q2n↑	SCC↑	D_λ↓	D_s↓	QNR↑	HQNR↑
(0.6, 0.2, 0.2)	4.55±0.56	3.95±0.54	0.928±0.082	0.979±0.009	0.045±0.022	0.068±0.023	0.890±0.022	0.889±0.021
(0.5, 0.4, 0.1)	4.52±0.63	3.87±0.44	0.932±0.067	0.980±0.012	0.043±0.023	0.062±0.021	0.898±0.019	0.899±0.024
(0.5, 0.3, 0.2)	**4.45±0.59**	**3.72±0.47**	**0.934±0.072**	**0.981±0.009**	**0.040±0.019**	**0.054±0.026**	**0.908±0.024**	**0.908±0.023**
(0.5, 0.2, 0.3)	4.47±0.57	3.79±0.48	0.933±0.071	0.980±0.008	0.045±0.017	0.058±0.028	0.900±0.020	0.899±0.022
(0.5, 0.1, 0.4)	4.57±0.57	3.96±0.46	0.929±0.068	0.979±0.009	0.044±0.021	0.066±0.024	0.893±0.024	0.893±0.023
(0.4, 0.3, 0.3)	4.56±0.57	3.94±0.43	0.930±0.063	0.980±0.011	0.043±0.017	0.065±0.025	0.895±0.025	0.896±0.024

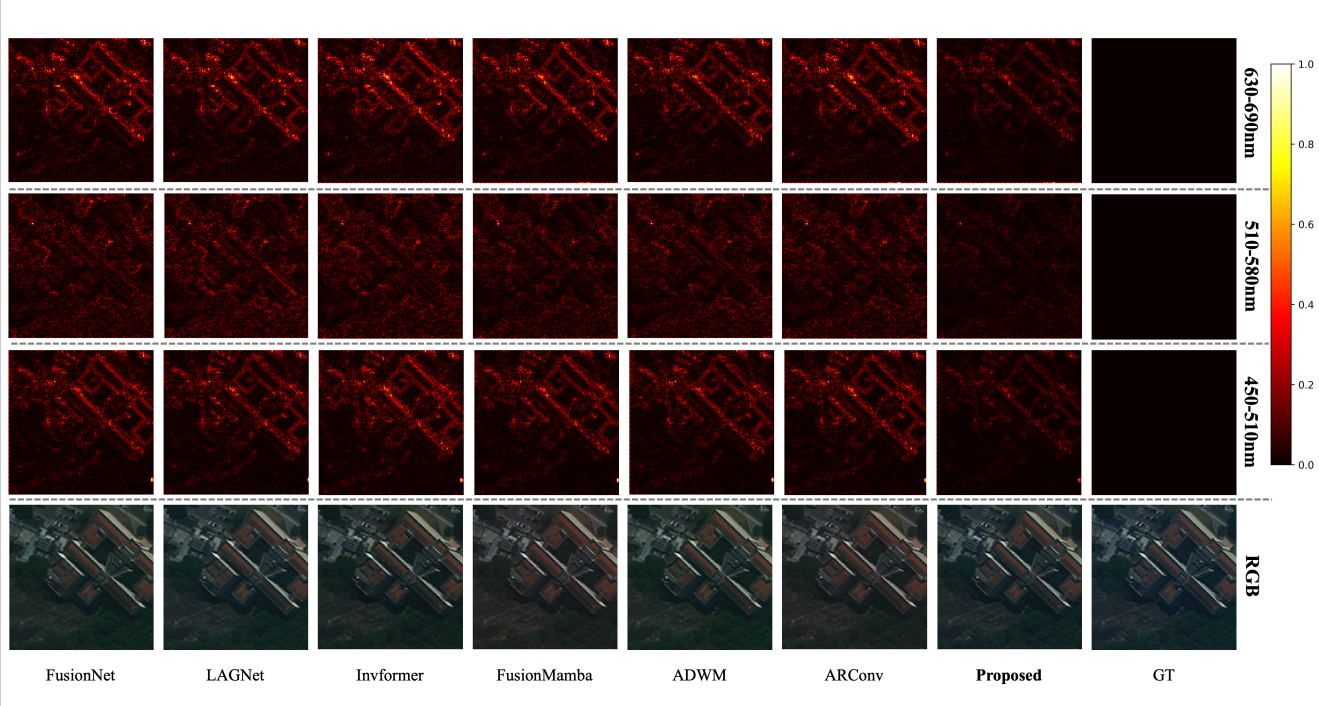

Figure 1. Qualitative comparison of different methods on sample 1 of WV3 dataset. Top three rows: reconstruction error maps; Bottom row: RGB outputs.

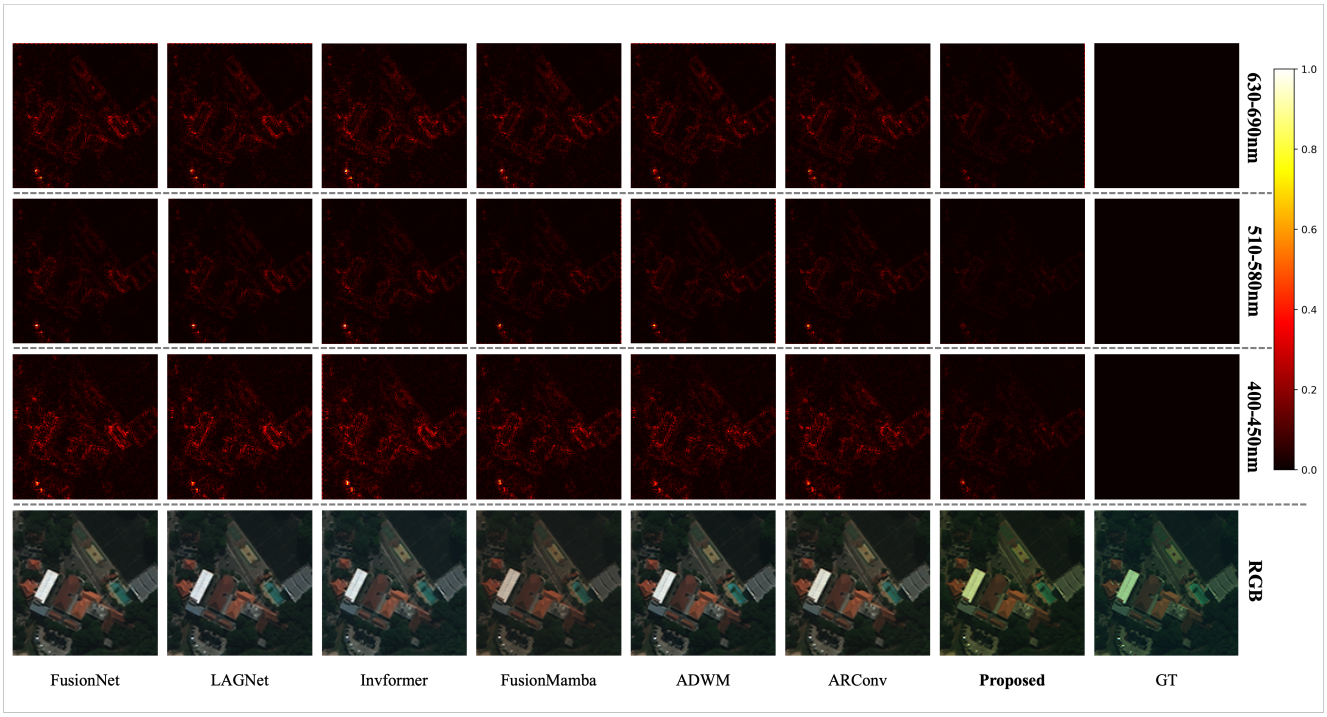

Figure 2. Qualitative comparison of different methods on sample 2 of WV3 Dataset. Top three rows: reconstruction error maps; Bottom row: RGB outputs.

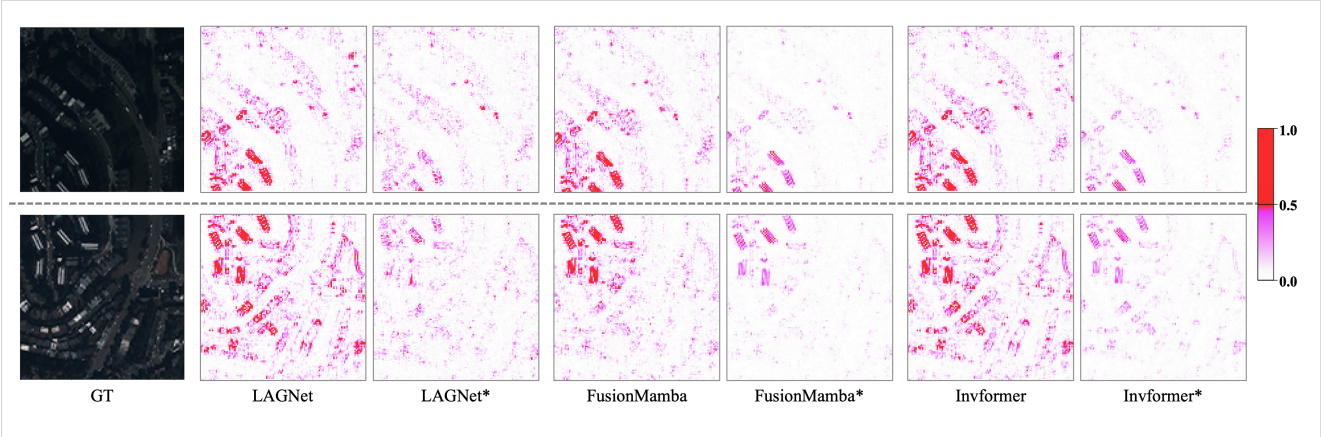

Figure 3. Reconstruction error comparison of different methods on the QB dataset. The suffix '*' denotes integration with our method.

Figure 4. Comparison of gradient contribution: L1 Loss versus HAL across training iterations. The darker the color in the gradient map, the greater the proportion of its gradient contribution. The training sample is from the GF2 dataset, where the spectral response range for gradient calculation is 520-590nm.

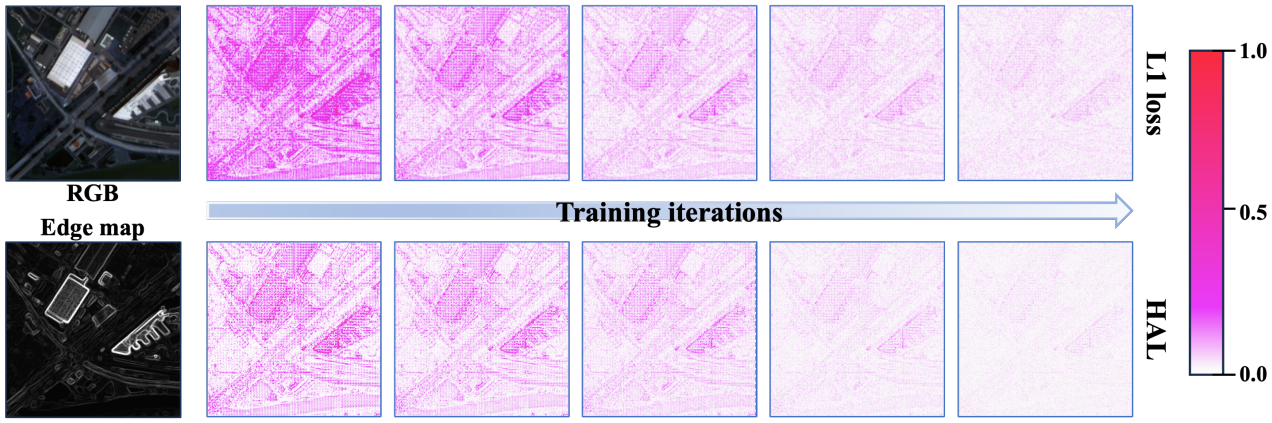

Figure 5. Comparison of gradient contribution: L1 Loss versus HAL across training iterations. The darker the color in the gradient map, the greater the proportion of its gradient contribution. The training sample is from the GF2 dataset, where the spectral response range for gradient calculation is 630-690nm.

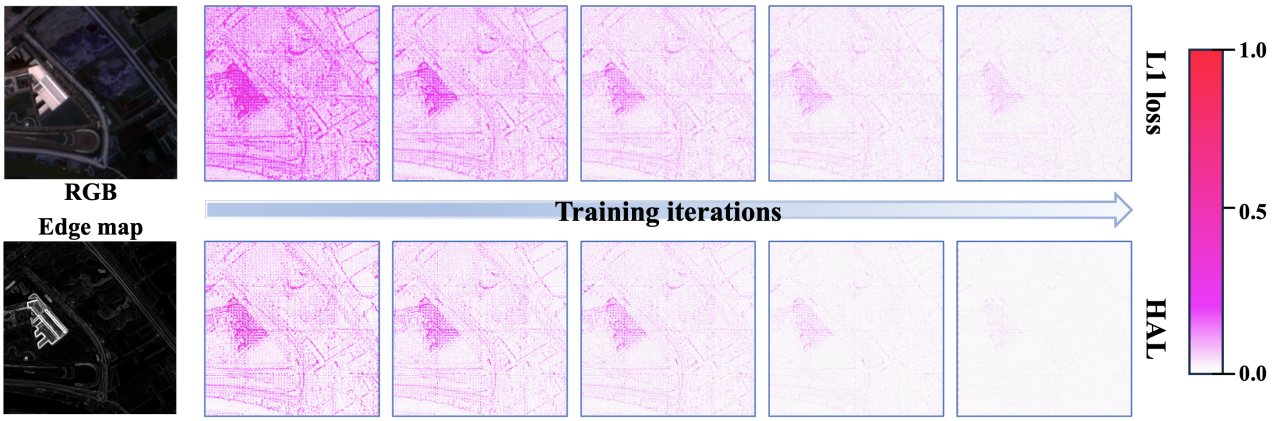

Figure 6. Comparison of gradient contribution: L1 Loss versus HAL across training iterations. The darker the color in the gradient map, the greater the proportion of its gradient contribution. The training sample is from the GF2 dataset, where the spectral response range for gradient calculation is 520-590nm.

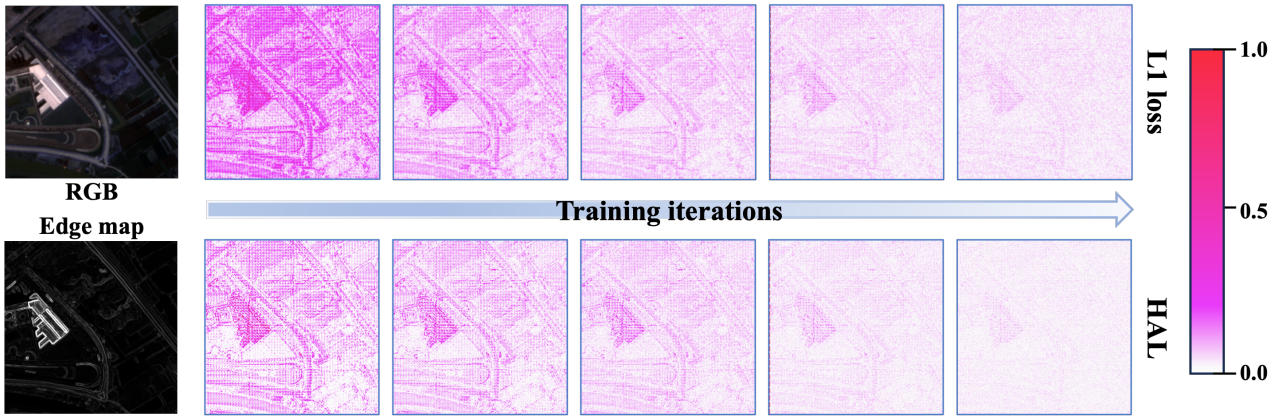

Figure 7. Comparison of gradient contribution: L1 Loss versus HAL across training iterations. The darker the color in the gradient map, the greater the proportion of its gradient contribution. The training sample is from the GF2 dataset, where the spectral response range for gradient calculation is 630-690nm.

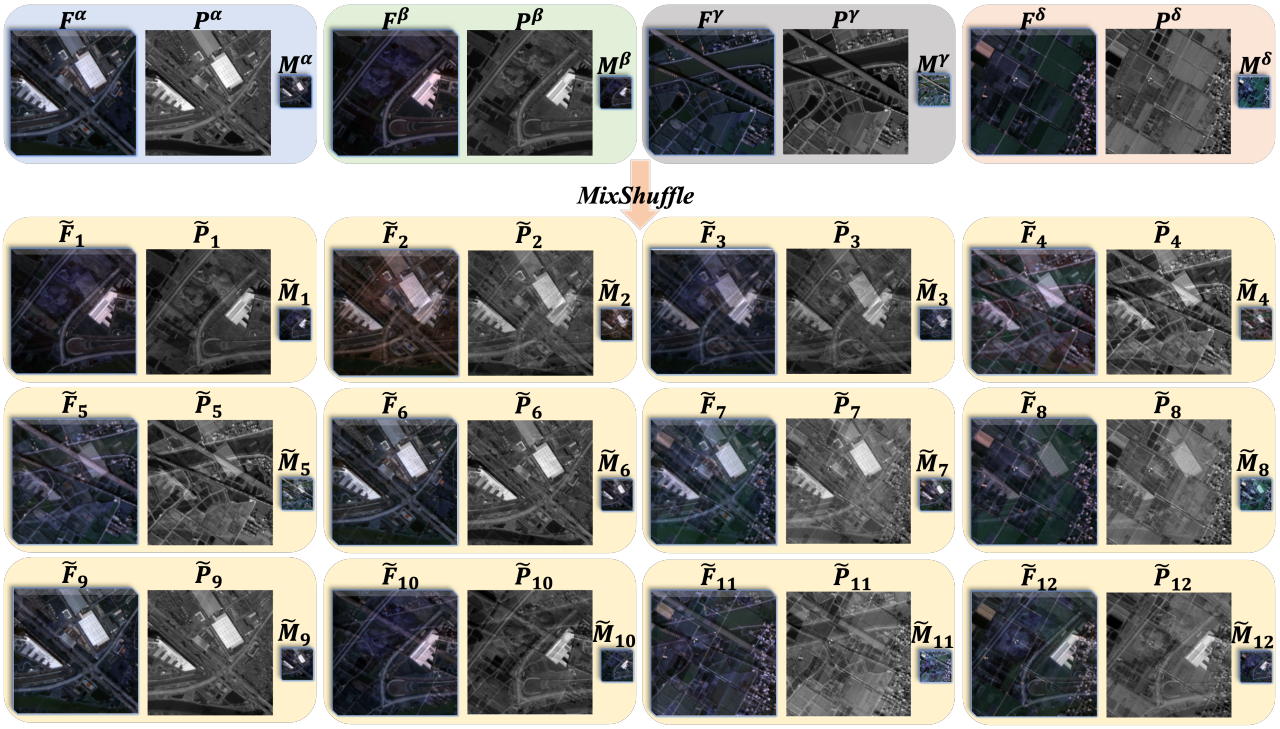

Figure 8. More examples of new training samples obtained through MixShuffle. The top row shows the original training samples; the bottom three rows display examples of new samples generated by MixShuffle.