Change the Base, Change the Game: Rethinking Pan-sharpening

Jeonghyeok Do *
KAIST
ehwjdgur0913@kaist.ac.kr

Geunhyuk Youk *
KAIST
rmsgurkjg@kaist.ac.kr

Munchurl Kim †
KAIST
mkimee@kaist.ac.kr

https://kaist-viclab.github.io/PanBase_site

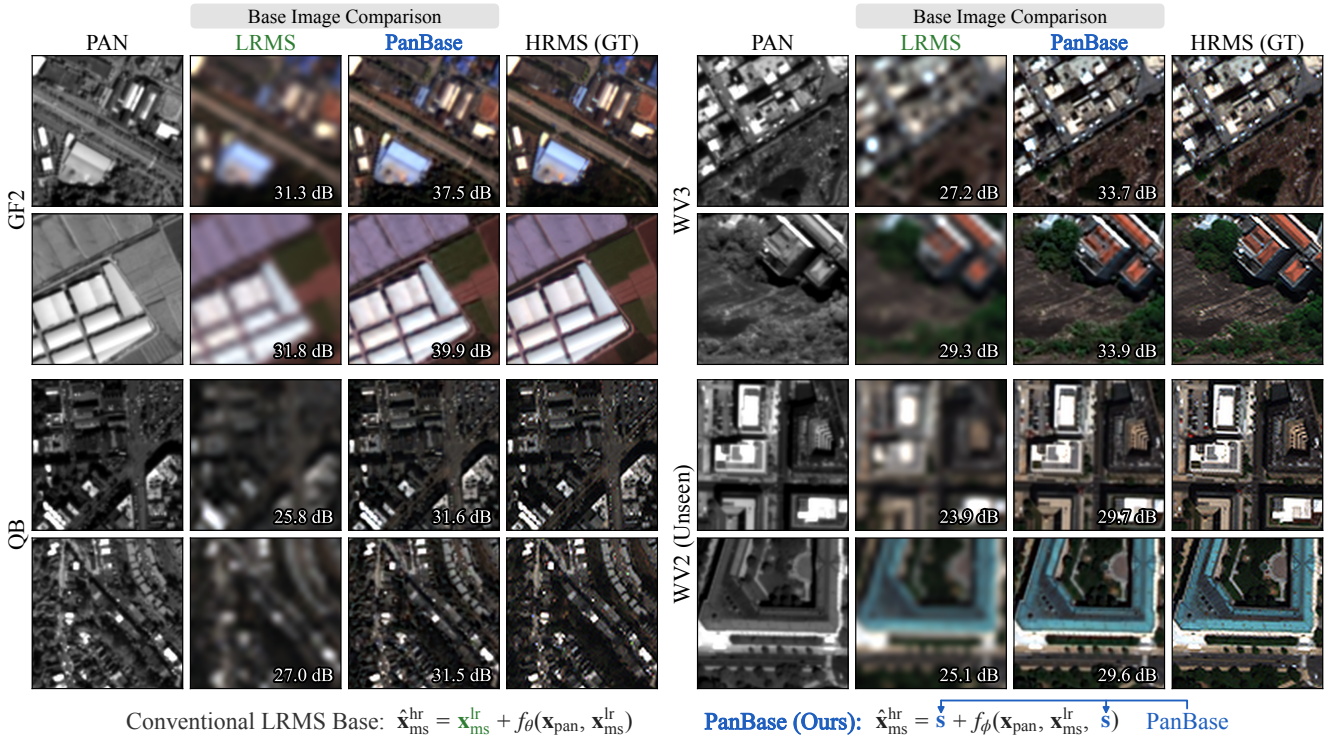

Figure 1. **Why start with a blurry base? Same backbone, a sharper base.** Examples from PanCollection span GaoFen-2 (GF2), QuickBird (QB), WorldView-3 (WV3), and WorldView-2 (WV2). The reported PSNR values compare interpolated LRMS and PanBase against the HRMS ground truth ($\mathbf{x}_{\text{ms}}^{\text{hr}}$) before any neural refinement. PanBase replaces the interpolated LRMS in the global skip and also serves as an additional backbone input. The final HRMS prediction ($\hat{\mathbf{x}}_{\text{ms}}^{\text{hr}}$) is obtained by adding the backbone's predicted residual to the base. For unseen WV2, PanBase reuses the coefficients calibrated on WV3 without refitting.

Abstract

Many pan-sharpening (PS) networks reconstruct high-resolution multispectral (HRMS) images by adding learned residuals to 4×-interpolated low-resolution multispectral (LRMS) inputs, incorporating panchromatic (PAN) detail only through these residual corrections. This raises a simple question: Why start with a blurry base when sharp spatial detail is already available? We propose ***PanBase****, a sharper multispectral starting point designed for more effective residual learning. Guided by physical sensor characteristics, PanBase combines PAN spatial detail with LRMS spectral information while enforcing consistency with the LRMS observations under sensor-specific blur. We construct this base without neural network training, estimating only nine coefficients per spectral band through closed-form calibration. Our PanBase can be easily integrated into existing backbones with a global residual skip through two simple changes: (i) replacing the skip base with PanBase and (ii) concatenating it as an additional input. Using each backbone's original training recipe and matched training budgets, PanBase improves most reduced-*

*These authors contributed equally.
†Corresponding author.

and full-resolution metrics on WorldView-3, QuickBird, and GaoFen-2, while achieving 4.4× faster training convergence on average. In zero-shot evaluation on WorldView-2, standalone PanBase achieves competitive reconstruction without target-sensor fitting or neural refinement.

1. Introduction

Residual learning is widely used in image restoration, including super-resolution [19], denoising [46], and deblurring [6]. In this approach, a global skip connection carries an initial estimate, or **base**, derived from the available observations, such as low-resolution, noisy, or blurred images. The network adds its predicted residual to this base to reconstruct the target. A base that captures more of the target structure leaves less for the network to recover, which can accelerate convergence and improve reconstruction quality.

Pan-sharpening (PS) offers a natural opportunity to build a more informative base because it reconstructs a high-resolution multispectral (HRMS) image from two complementary observations: a panchromatic (PAN) image with fine spatial detail and a low-resolution multispectral (LRMS) image with band-specific spectral information. (i) Starting from PAN, PS resembles colorization: PAN provides the spatial structure, and LRMS guides the recovery of the spectral bands. (ii) Starting from LRMS, PS resembles $4\times$ super-resolution: LRMS provides the spectral information, and PAN guides the recovery of fine spatial detail. In either view, one observation provides the starting point and the other guides the recovery of what is missing. This raises a natural question: *Why not build a better base from both observations before training the residual network?*

Recent PS methods have improved residual prediction through adaptive convolutions [11, 37], modality-aware attention [10], state-space models [28], and diffusion [20]. Yet many residual networks still use interpolated LRMS as the base in their global skip connection [45]. Although the residual branch receives PAN, the base does not directly incorporate its fine spatial detail. This leaves an opportunity to improve the starting point itself using detail that is already available in PAN.

In this paper, we propose **PanBase**, a sharper multispectral base constructed from PAN and LRMS before backbone training. Guided by the sensor's physical characteristics, PanBase incorporates spatial detail from PAN while ensuring that the resulting base reproduces the observed LRMS after sensor-specific blur and downsampling. Constructing the base requires no neural network training: closed-form ridge regression [16] calibrates just nine coefficients per spectral band, giving 36 and 72 coefficients for four- and eight-band images, respectively. Fig. 1 shows that PanBase provides a sharper starting point even before neural refinement. We integrate PanBase with only two changes: (i) replacing interpolated LRMS with PanBase in the global skip and (ii) concatenating PanBase with the original inputs. This design is compatible with diverse backbones that use a global image-space residual connection, including convolution-, attention-, state-space-, and diffusion-based models.

On PanCollection datasets, adding PanBase improves most reduced-resolution (RR) and full-resolution (FR) metrics on WorldView-3, QuickBird, and GaoFen-2 under matched training budgets. Across all four sensors, including WorldView-2, it improves RR PSNR in 39 of 40 matched comparisons, with gains of up to 6.93 dB. PanBase alone outperforms all compared classical methods in all metrics across the four sensors. On unseen WorldView-2, PanBase reuses the 72 coefficients calibrated on WorldView-3 and achieves 30.12 dB PSNR, surpassing all original backbones in the main comparison without target-sensor fitting or neural refinement.

PanBase also accelerates training convergence by a factor of $4.4\times$ on average, while keeping the cost of base construction modest. For four-band data, the one-time calibration takes 88 s to load 6,000 training patches and accumulate calibration statistics, followed by 16 ms for the coefficient solves. Once the coefficients and sensor operators are in place, constructing a four-band 256×256 base takes 0.19 s on CPU.

Our contributions are threefold:

- We rethink residual pan-sharpening by changing its starting point: **PanBase** incorporates the sharp spatial detail already available in PAN into the multispectral base before residual learning, while preserving consistency with the observed LRMS.
- We calibrate just nine coefficients per spectral band in closed form and integrate PanBase into existing residual backbones by replacing the global skip base and concatenating PanBase with the original inputs.
- Across ten backbones and four sensors, matched comparisons show RR PSNR gains of up to 6.93 dB and faster training, while standalone PanBase transfers to an unseen sensor without recalibration.

2. Related Work

2.1. Pan-sharpening (PS)

Traditional methods. Traditional PS methods include component substitution (CS), multiresolution analysis (MRA), and variational optimization (VO) [32, 36]. CS transfers spatial structure by substituting an MS-derived intensity component with PAN [3, 7, 23, 31]. MRA extracts PAN detail through multiscale decomposition and injects it into upsampled MS using band-dependent gains [2, 5, 25, 29, 34]. VO reconstructs HRMS by optimizing

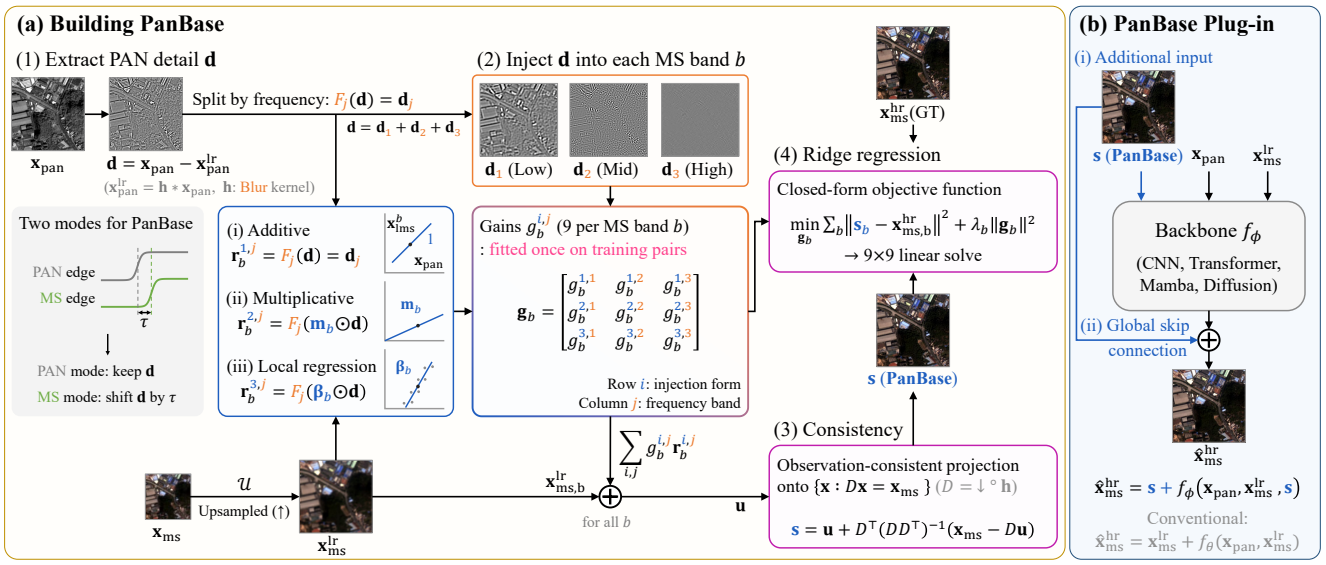

Figure 2. **PanBase: construct the base, then learn its correction.** (a) Three PAN-detail forms and three frequency bands yield nine coefficients per MS band, calibrated by ridge regression and held fixed. Projection enforces LRMS consistency under the sensor model. (b) The base replaces the global skip and is concatenated with the backbone inputs; added input-channel weights are zero-initialized.

observation fidelity together with spatial or spectral priors [26, 27, 33, 41]. While these methods directly reconstruct HRMS, PanBase uses the sensor's physical imaging model to combine PAN spatial detail with LRMS spectral information into an observation-consistent starting point for residual learning.

Learning-based methods. Early CNN-based methods, such as PanNet, introduced residual learning for spatial–spectral reconstruction [45]. Subsequent designs improve feature aggregation through adaptive convolutions in LAGConv [18] and encourage complementary PAN–MS representations through mutual information learning in MutInf [47]. HFIN extends fusion to hierarchical spatial–frequency representations [30], while PAN-Crafter addresses PAN–MS misalignment through modality-aware attention [10]. Recent methods further explore two-step spatial–spectral enhancement in TS-Pan [21] and efficient binary reconstruction in BNNPan [17]. State-space models also support long-range feature interaction [28]. Beyond architectural advances, MDCUN embeds degradation models into deep unfolding [43], while PINO incorporates sensor imaging physics into learned fusion [22]. PanFlowNet further explores probabilistic reconstruction through conditional normalizing flows [44]. Diffusion-based methods extend generative reconstruction through text conditioning [42], uncertainty-aware distillation [20], and few-step sampling [12]. PanBase complements these approaches with a sharper, observation-consistent base constructed in closed form and can be integrated into any backbone with a global image-space residual connection.

2.2. Residual Learning for Image Restoration

Residual learning preserves an initial estimate and predicts its correction, as in super-resolution [19], denoising [46], and deblurring [6]. The initial estimate defines the residual target and determines which structures remain for the network to recover. In PS, residual networks commonly use upsampled LRMS as the base, introducing PAN spatial detail through the learned correction. PanNet learns spatial corrections in the high-pass domain while preserving spectral information through an upsampled-MS skip [45]. DiCNN directly models the injected MS detail [14], while CS-Net, MRA-Net, and FusionNet connect classical injection formulations with learned nonlinear detail prediction [8]. Related efforts also improve initial estimates through PAN-guided learned upsampling in PGCU [48] or use a frozen network-generated spectral base for residual diffusion [40]. PanBase instead constructs the residual reference through closed-form calibration with explicit LRMS consistency, without requiring a learned upsampler or pre-enhancement network.

3. PanBase

PanBase aims to construct a sharper, observation-consistent multispectral base from PAN and LRMS that serves as the starting point for residual learning. Let $\mathbf{x}_{pan} \in \mathbb{R}^{H \times W}$, $\mathbf{x}_{ms} \in \mathbb{R}^{C \times h \times w}$, and $\mathbf{x}^{hr}_{ms} \in \mathbb{R}^{C \times H \times W}$ denote the PAN image, LRMS image, and HRMS target, respectively, with scale ratio $r = H/h = W/w$. We write $\mathbf{x}^{lr}_{ms} = \mathcal{U}(\mathbf{x}_{ms})$ for upsampled LRMS and $\mathbf{s} \in \mathbb{R}^{C \times H \times W}$ for PanBase.

Fig. 2 summarizes two stages: constructing the base and

learning a residual correction around it. Starting from $\mathbf{x}_{\mathrm{ms}}^{\mathrm{lr}}$, we inject a weighted combination of PAN detail components adapted to each spectral band and frequency range (Sec. 3.1). The initial base is then projected to reproduce the observed LRMS under sensor-specific blur and downsampling (Sec. 3.2). The resulting PanBase serves as both the global residual skip and an additional backbone input (Sec. 3.3). Its coefficients are calibrated once on training pairs and held fixed during backbone training and inference.

3.1. PAN Detail Injection

We first extract high-frequency detail from the PAN image. Let $\mathbf{h}_b$ denote the sensor blur kernel for MS band b, and let $\bar{\mathbf{h}}$ denote the mean blur kernel across MS bands. The blurred PAN image and its detail map are given by

$$\mathbf{x}_{\mathrm{pan}}^{\mathrm{lr}} = \bar{\mathbf{h}} * \mathbf{x}_{\mathrm{pan}}, \qquad \mathbf{d} = \mathbf{x}_{\mathrm{pan}} - \mathbf{x}_{\mathrm{pan}}^{\mathrm{lr}}, \tag{1}$$

where $*$ denotes spatial convolution. Both quantities remain on the $H \times W$ PAN grid. We consider two variants, $\mathbf{s}_{\mathrm{pan}}$ and $\mathbf{s}_{\mathrm{ms}}$, that differ in the spatial alignment of the PAN detail. The PAN variant retains $\mathbf{d}$ in its native geometry, whereas the MS variant shifts it by an input-estimated translation $\boldsymbol{\tau}$ to align with the MS geometry. Since all subsequent steps are identical, we use $\mathbf{d}$ for the detail map of either variant throughout the following formulation.

PAN–MS relation modeling. To capture band-dependent PAN–MS relationships, we define two relation maps: an intensity ratio map $\mathbf{m}_b$ and a local regression map $\boldsymbol{\beta}_b$ [2]:

$$\mathbf{m}_b = \frac{\mathbf{x}_{\mathrm{ms},b}^{\mathrm{lr}}}{\mathbf{x}_{\mathrm{pan}}^{\mathrm{lr}}}, \qquad \boldsymbol{\beta}_b = \mathcal{U}\left[\frac{\mathrm{cov}_w(\mathbf{x}_{\mathrm{ms},b}, \mathbf{p})}{\mathrm{var}_w(\mathbf{p})}\right], \tag{2}$$

where $\mathbf{p} = \downarrow_r (\mathbf{x}_{\mathrm{pan}}^{\mathrm{lr}})$ is PAN degraded to the LRMS grid, w is a 7×7 local window, and $\mathcal{U}$ upsamples the regression map to the HR grid. The regression map measures the local change in each MS band with respect to degraded PAN, allowing both positive and negative responses. Both maps are computed from each PAN–LRMS input pair.

Using these relation maps, we construct three complementary PAN detail injection forms $\mathbf{r}_b^i$, $i \in \{1, 2, 3\}$, for each MS band b:

$$\mathbf{r}_b^1 = \mathbf{d}, \qquad \mathbf{r}_b^2 = \mathbf{m}_b \odot \mathbf{d}, \qquad \mathbf{r}_b^3 = \boldsymbol{\beta}_b \odot \mathbf{d}, \tag{3}$$

where the three forms correspond to additive, multiplicative, and local regression injection, respectively, and multiplication $\odot$ and division are performed pointwise.

Frequency-dependent representation. Applying a single scalar weight to an injection form scales all its frequencies equally, limiting independent control over detail at different spatial scales and noise. We therefore decompose each $\mathbf{r}_b^i$ into three smooth frequency bands:

$$\mathbf{r}_b^{i,j} = F_j(\mathbf{r}_b^i) = \mathcal{F}^{-1}\left[\alpha_j \odot \mathcal{F}(\mathbf{r}_b^i)\right], \quad j \in \{1, 2, 3\}, \tag{4}$$

where F_j extracts the j-th frequency band, $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote the Fourier transform and its inverse, and α_j are fixed, smooth frequency weights for the low-, middle-, and high-frequency bands, respectively. The weights α_j sum to one at every Fourier frequency, ensuring $\sum_{j=1}^{3} \mathbf{r}_b^{i,j} = \mathbf{r}_b^i$. The three injection forms and three frequency bands yield nine PAN detail components $\{\mathbf{r}_b^{i,j}\}$ per MS band b.

Weighted PAN detail injection. We assign a scalar coefficient $g_b^{i,j}$ to each component $\mathbf{r}_b^{i,j}$, indexed by injection form i and frequency band j, and collect the nine coefficients in the ordered vector $\mathbf{g}_b = [g_b^{1,1}, g_b^{1,2}, \ldots, g_b^{3,3}]^\top \in \mathbb{R}^9$. Adding their weighted combination to upsampled LRMS gives an initial base parameterized by $\mathbf{g}_b$:

$$\mathbf{u}_b(\mathbf{g}_b) = \mathbf{x}_{\mathrm{ms},b}^{\mathrm{lr}} + \sum_{i=1}^{3}\sum_{j=1}^{3} g_b^{i,j} \mathbf{r}_b^{i,j}. \tag{5}$$

Stacking the bandwise estimates yields the initial multispectral base $\mathbf{u}$.

3.2. Observation-Consistent Calibration

Throughout this subsection, we vectorize each image band and reuse the same symbols for the resulting column vectors. The LRMS observation is $\mathbf{x}_{\mathrm{ms},b} \in \mathbb{R}^{hw}$. All HR image quantities, including upsampled LRMS $\mathbf{x}_{\mathrm{ms},b}^{\mathrm{lr}}$, the HRMS target $\mathbf{x}_{\mathrm{ms},b}^{\mathrm{hr}}$, detail components $\mathbf{r}_b^{i,j}$, and bases $\mathbf{u}_b$ and $\mathbf{s}_b$, are vectors in $\mathbb{R}^{HW}$.

Observation-consistent projection. Detail injection alone does not guarantee that the initial base reproduces the observed LRMS after sensor-specific blur and downsampling. Let $\mathbf{D}_b \in \mathbb{R}^{hw \times HW}$ denote the matrix representation of the sensor degradation operator $\downarrow_r \circ (\mathbf{h}_b * \cdot)$. We define the LRMS-consistent set as

$$\mathcal{C}_b = \left\{\mathbf{x} \in \mathbb{R}^{HW} \,\middle|\, \mathbf{D}_b\mathbf{x} = \mathbf{x}_{\mathrm{ms},b}\right\}. \tag{6}$$

For full-row-rank $\mathbf{D}_b$, the Euclidean projection onto $\mathcal{C}_b$ is

$$\begin{aligned} \Pi_b(\mathbf{u}) &= \arg\min_{\mathbf{x} \in \mathcal{C}_b} \tfrac{1}{2}\|\mathbf{x} - \mathbf{u}\|_2^2 \\ &= \mathbf{u} + \mathbf{D}_b^\dagger(\mathbf{x}_{\mathrm{ms},b} - \mathbf{D}_b\mathbf{u}), \end{aligned} \tag{7}$$

where $\mathbf{D}_b^\dagger = \mathbf{D}_b^\top(\mathbf{D}_b\mathbf{D}_b^\top)^{-1}$. Applying this projection to the initial base yields PanBase:

$$\mathbf{s}_b(\mathbf{g}_b) = \Pi_b(\mathbf{u}_b(\mathbf{g}_b)). \tag{8}$$

By construction, $\mathbf{s}_b(\mathbf{g}_b) \in \mathcal{C}_b$, so $\mathbf{D}_b\mathbf{s}_b(\mathbf{g}_b) = \mathbf{x}_{\mathrm{ms},b}$ for any coefficient vector $\mathbf{g}_b$, up to numerical solver accuracy. Thus, calibration adjusts PAN detail while preserving agreement with the LRMS observation under the prescribed sensor model.

Closed-form coefficient calibration. For fixed PAN and LRMS inputs, substituting Eq. (5) into Eq. (8) gives the affine form

$$\mathbf{s}_b(\mathbf{g}_b) = \mathbf{s}_{0,b} + \mathbf{A}_b\mathbf{g}_b, \tag{9}$$

Table 1. **PanBase as a plug-in for 10 backbones.** Highlighted rows use PanBase $\mathbf{s}_{\text{pan}}$ as the global residual base and an additional input under matched training. Green/red deltas indicate gains/drops over each baseline; **bold**/underline denote the best/second-best PSNR and HQNR. WV2† uses WV3-trained models without fine-tuning. Speed-up compares iterations to reach the original best score: per-backbone medians over reached WV3/QB/GF2 targets, summarized by their geometric mean.

Method	Venue	RR PSNR↑ (dB)				FR HQNR↑				Training speed-up↑
		WV3	QB	GF2	WV2†	WV3	QB	GF2	WV2†	
PanNet [45]	ICCV'17	34.29	33.83	36.87	28.51	0.934	0.843	0.915	**0.968**	–
+ PanBase	–	34.96 +0.67	34.20 +0.37	37.98 +1.10	29.10 +0.59	0.967 +0.034	0.857 +0.015	0.927 +0.012	0.958 −0.010	60.0×
LAGConv [18]	AAAI'22	37.22	37.96	42.47	28.71	0.944	0.923	0.915	0.949	–
+ PanBase	–	37.34 +0.13	38.11 +0.15	42.82 +0.35	29.17 +0.46	0.958 +0.014	0.911 −0.012	0.949 +0.034	0.903 −0.046	2.1×
MutInf [47]	CVPR'22	35.77	36.64	36.65	28.87	0.913	0.876	0.943	0.902	–
+ PanBase	–	37.05 +1.28	38.33 +1.69	43.59 +6.93	29.01 +0.14	0.967 +0.055	0.905 +0.029	0.973 +0.031	0.940 +0.038	9.2×
MDCUN [43]	CVPR'22	36.95	35.99	42.04	28.96	0.961	0.914	0.953	0.938	–
+ PanBase	–	37.12 +0.17	37.66 +1.67	42.57 +0.53	30.05 +1.09	0.967 +0.006	**0.943** +0.029	0.964 +0.011	0.961 +0.023	4.3×
PanFlowNet [44]	ICCV'23	35.76	36.23	38.56	27.78	0.918	0.853	0.904	0.869	–
+ PanBase	–	36.53 +0.76	37.87 +1.65	40.12 +1.55	28.19 +0.41	0.954 +0.036	0.831 −0.022	0.928 +0.024	0.936 +0.067	5.0×
HFIN [30]	CVPR'24	36.53	35.17	38.08	28.39	0.917	0.796	0.909	0.903	–
+ PanBase	–	37.26 +0.74	37.42 +2.25	40.95 +2.87	29.47 +1.08	0.965 +0.049	0.924 +0.128	0.920 +0.011	0.952 +0.048	6.1×
BNNPan [17]	CVPR'25	36.73	37.38	42.12	27.35	0.936	0.856	0.927	0.755	–
+ PanBase	–	37.05 +0.32	37.71 +0.33	42.69 +0.57	29.69 +2.34	**0.969** +0.034	0.880 +0.024	0.942 +0.015	0.949 +0.194	1.2×
PAN-Crafter [10]	ICCV'25	37.48	35.75	44.33	29.04	0.964	0.826	0.971	0.949	–
+ PanBase	–	37.54 +0.06	37.74 +1.99	**44.80** +0.48	29.90 +0.86	0.962 −0.002	0.846 +0.020	0.979 +0.008	0.937 −0.012	1.7×
PINO [22]	NeurIPS'25	36.42	37.37	42.48	29.76	0.959	0.925	0.979	0.946	–
+ PanBase	–	37.26 +0.84	38.12 +0.75	44.20 +1.72	**30.14** +0.38	0.961 +0.002	0.935 +0.009	**0.986** +0.007	0.953 +0.007	3.8×
TS-Pan [21]	TGRS'26	**37.90**	38.53	41.94	29.59	0.946	0.743	0.944	0.920	–
+ PanBase	–	37.51 −0.40	**38.54** +0.01	44.34 +2.40	30.10 +0.52	0.955 +0.010	0.942 +0.198	0.931 −0.013	0.930 +0.010	2.5×
Mean change ($\mathbf{s}_{\text{pan}}$)		+0.46	+1.09	+1.85	+0.79	+0.024	+0.042	+0.014	+0.032	4.4×

where $\mathbf{s}_{0,b} = \Pi_b(\mathbf{x}^{\text{lr}}_{\text{ms},b})$ and $\mathbf{A}_b \in \mathbb{R}^{HW\times 9}$ contains the nine columns $(\mathbf{I}_{HW} - \mathbf{D}_b^{\dagger}\mathbf{D}_b)\mathbf{r}_b^{i,j}$, ordered consistently with $\mathbf{g}_b$. For training pair $k \in \{1, \dots, K\}$, define the target correction as $\mathbf{e}_b^k = \mathbf{x}^{\text{hr},k}_{\text{ms},b} - \mathbf{s}^k_{0,b}$. We calibrate the nine coefficients jointly across training pairs by ridge regression:

$$\mathbf{g}_b^{\star} = \arg\min_{\mathbf{g}_b} \sum_k \frac{\|\mathbf{A}_b^k \mathbf{g}_b - \mathbf{e}_b^k\|_2^2}{HW} + \lambda_b \|\mathbf{g}_b\|_2^2. \tag{10}$$

The closed-form solution is $\mathbf{g}_b^{\star} = (\mathbf{H}_b + \lambda_b \mathbf{I}_9)^{-1}\mathbf{c}_b$, where

$$\mathbf{H}_b = \sum_k \frac{(\mathbf{A}_b^k)^{\top}\mathbf{A}_b^k}{HW}, \quad \mathbf{c}_b = \sum_k \frac{(\mathbf{A}_b^k)^{\top}\mathbf{e}_b^k}{HW}. \tag{11}$$

We use $\lambda_b = 10^{-2}\,\text{tr}(\mathbf{H}_b)/9$ and set $\mathbf{g}_b^{\star} = \mathbf{0}$ when $\mathbf{H}_b = \mathbf{0}$. After accumulating the training statistics, coefficient calibration requires only one 9×9 linear solve per MS band. This supervised calibration uses HRMS targets without neural network training and produces $9C$ coefficients that remain fixed during backbone training and inference. At inference, PanBase is generated as $\mathbf{s}_b = \Pi_b(\mathbf{u}_b(\mathbf{g}_b^{\star}))$ using only PAN, LRMS, the prescribed sensor operators, and the calibrated coefficients.

3.3. Residual Backbone Integration

We replace the global residual base with PanBase $\mathbf{s}$ and concatenate it with the original backbone inputs:

$$\widehat{\mathbf{x}}^{\text{hr}}_{\text{ms}} = \mathbf{s} + f_{\phi}\left(\mathbf{x}_{\text{pan}}, \mathbf{x}^{\text{lr}}_{\text{ms}}, \mathbf{s}\right). \tag{12}$$

(i) Replacing the interpolated-LRMS skip changes the residual target to $\mathbf{x}^{\text{hr}}_{\text{ms}} - \mathbf{s}$. (ii) Supplying $\mathbf{s}$ to the residual branch also exposes the estimate that the backbone must correct. We expand only the input layer to accommodate PanBase, keeping the remaining architecture unchanged. This integration applies to any backbone with a global image-space residual connection.

4. Experiments

4.1. Datasets

We evaluate PanBase on four pan-sharpening datasets from PanCollection [9]: WorldView-3 (WV3), QuickBird (QB), GaoFen-2 (GF2), and WorldView-2 (WV2). For training on WV3, QB, and GF2, we use PAN patches with $H = W = 64$ and LRMS patches with $h = w = 16$, giving $r = 4$. The number of MS bands is $C = 4$ for QB and GF2 and $C = 8$ for WV3 and WV2. Test PAN images have spatial sizes of 256×256 at reduced resolution (RR) and 512×512 at full resolution (FR). We reserve WV2 exclusively for unseen-sensor evaluation, reusing the WV3-calibrated injection coefficients without target-sensor recalibration or backbone fine-tuning.

4.2. Implementation Details

We use each backbone's official implementation and training recipe, retaining the optimizer, learning rate, batch size,

Table 2. **Classical pan-sharpening versus standalone PanBase.** All methods operate without neural networks; PanBase coefficients are calibrated once using HRMS training pairs. CS/MRA/VO denote component substitution, multiresolution analysis, and variational optimization. On WV2, PanBase reuses WV3 coefficients with the WV2 sensor operator.

Type	Method	WV3			QB			GF2			WV2 (Unseen)		
		PSNR↑	SAM↓	HQNR↑	PSNR↑	SAM↓	HQNR↑	PSNR↑	SAM↓	HQNR↑	PSNR↑	SAM↓	HQNR↑
–	LRMS↑ (interpolated) [1]	27.12	5.80	0.909	28.06	8.44	0.831	32.08	1.82	0.905	25.38	6.53	0.922
CS	BT-H [23]	31.22	4.92	0.891	32.09	7.23	0.685	35.80	1.56	0.832	28.90	5.90	0.879
	BDSD-PC [31]	30.96	5.43	0.895	31.89	8.09	0.750	35.10	1.68	0.829	29.04	6.14	0.888
	GSA [3]	30.78	5.37	0.896	31.75	8.10	0.687	34.76	1.72	0.792	28.51	6.25	0.877
	PRACS [7]	29.95	5.58	0.935	31.00	8.19	0.781	35.32	1.69	0.865	27.24	6.32	0.941
MRA	AWLP [25]	31.21	5.23	0.938	31.89	8.19	0.837	34.45	1.94	0.857	28.87	6.07	0.917
	MTF-GLP-FS [34]	30.91	5.32	0.929	32.02	7.79	0.830	35.43	1.66	0.853	28.59	6.19	0.911
	MTF-GLP-HPM [2, 5]	31.38	5.13	0.920	32.24	7.79	0.826	34.71	1.65	0.841	29.10	6.06	0.904
	MF [29]	30.56	5.26	0.919	31.47	7.93	0.831	34.71	1.64	0.830	28.41	6.13	0.898
VO	TV [26]	30.48	5.69	0.952	31.59	7.51	0.875	35.04	1.91	0.881	28.25	6.03	0.954
	FE-HPM [33]	30.87	5.23	0.923	31.71	7.95	0.826	35.39	1.61	0.822	28.71	6.10	0.902
	PWMBF [27]	30.07	5.89	0.900	31.00	8.87	0.782	33.87	1.96	0.881	27.60	7.08	0.839
	LRTCFPan [41]	31.74	4.74	0.942	32.57	7.19	0.926	37.53	1.30	0.907	29.38	5.62	0.935
Ours	**PanBase** ($\mathbf{s}_{\text{pan}}$)	32.28	4.65	**0.963**	33.10	6.56	**0.944**	38.31	1.21	0.965	30.12	5.49	**0.959**
	PanBase ($\mathbf{s}_{\text{ms}}$)	**33.88**	**4.62**	0.915	**36.05**	**6.11**	0.703	**39.86**	**1.09**	**0.971**	**30.34**	**5.48**	0.948

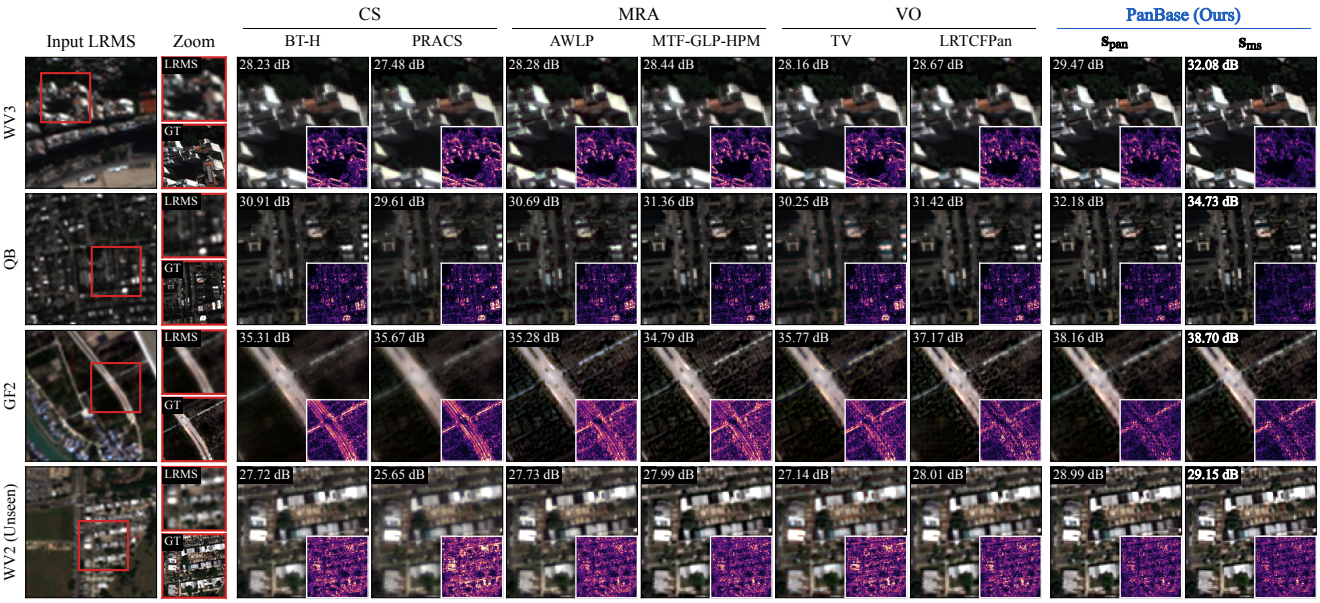

Figure 3. **Standalone PanBase versus classical fusion at RR.** PAN- and MS-aligned PanBase are compared with CS/MRA/VO methods without neural refinement. Red boxes mark enlarged regions; scores are PSNR (dB), and insets show reconstruction errors. Unseen WV2 reuses WV3-calibrated coefficients.

loss, and schedule. The only changes are replacing the global residual base with PanBase, concatenating it with the original inputs, and expanding the input layer.

Experiments use up to four NVIDIA RTX A6000 GPUs and an Intel Core i9-13900K CPU. For each training dataset, we randomly sample $K = 6{,}000$ patches to calibrate the injection coefficients on the GPU. For four-band (eight-band) data, patch loading and statistics accumulation take 88 s (167 s), followed by 16 ms (31 ms) for the ridge solves. After calibration, generating PanBase on the CPU takes 0.19 s (0.80 s) per image at batch size one.

RR evaluation. Against HRMS references $\mathbf{x}_{\text{ms}}^{\text{hr}}$, PSNR measures reconstruction accuracy and SAM measures spectral angular error; higher PSNR and lower SAM are better.

FR evaluation. Without HRMS references, HQNR [36] measures spectral and spatial consistency with observed MS and PAN images; higher is better. Additional RR metrics (SSIM [39], ERGAS, SCC, Q_4/Q_8 [13]) and FR measures

Table 3. **Ablation of alternative residual bases.** Alternative bases share the same skip replacement, input concatenation, and training.

Base	PanNet [45], WV3		PanNet [45], GF2		LAGConv [18], WV3		LAGConv [18], GF2	
	PSNR↑	HQNR↑	PSNR↑	HQNR↑	PSNR↑	HQNR↑	PSNR↑	HQNR↑
LRMS↑ (Original)	34.29	0.934	36.87	0.915	37.22	0.944	42.47	0.915
Π(LRMS↑)	34.75 +0.46	0.956 +0.022	37.62 +0.74	0.915 -0.000	**37.35** +0.13	0.956 +0.012	**42.90** +0.43	0.918 +0.004
GSA [3] (CS)	34.30 +0.01	0.915 −0.019	35.69 −1.19	0.839 −0.076	37.14 −0.07	0.955 +0.011	42.01 −0.46	0.908 −0.006
MTF-GLP-HPM [2, 5] (MRA)	34.81 +0.52	0.931 −0.003	36.92 +0.05	0.885 −0.031	37.17 −0.05	0.944 -0.000	42.36 −0.10	0.921 +0.007
LRTCFPan [41] (VO)	34.81 +0.52	0.942 +0.009	37.80 +0.93	0.904 −0.011	37.24 +0.02	0.956 +0.013	42.24 −0.23	0.948 +0.034
PanBase ($\mathbf{s}_{\text{pan}}$)	**34.96** +0.67	**0.967** +0.034	**37.98** +1.10	**0.927** +0.012	37.34 +0.13	**0.958** +0.014	42.82 +0.35	**0.949** +0.034

Table 4. **Ablation of PanBase construction.** Network-free bases add components cumulatively, with coefficients recalibrated per row and projection applied after detail injection. Coeff. counts calibrated scalars for C MS bands; the highlighted row is the complete $\mathbf{s}_{\text{pan}}$. PSNR/SAM use RR and HQNR uses FR. WV2 reuses WV3 coefficients with its own sensor operator.

Configuration	Coeff.	WV3			QB			GF2			WV2 (Unseen)		
		PSNR↑	SAM↓	HQNR↑	PSNR↑	SAM↓	HQNR↑	PSNR↑	SAM↓	HQNR↑	PSNR↑	SAM↓	HQNR↑
Interpolated LRMS ($\mathbf{x}^{\text{lr}}_{\text{ms},b}$)	0	27.12	5.80	0.909	28.06	8.44	0.831	32.08	1.82	0.905	25.38	6.53	0.922
+ Projection ($\Pi_b(\cdot)$)	0	27.99	5.44	0.927	29.09	7.86	0.822	34.21	1.47	0.951	26.20	6.05	0.901
+ Additive detail ($\mathbf{r}^1_b = \mathbf{d}$)	C	31.60	5.26	0.964	32.77	7.23	0.928	37.72	1.29	0.957	29.38	5.66	0.954
+ Multiplicative detail ($\mathbf{r}^2_b = \mathbf{m}_b \odot \mathbf{d}$)	$2C$	32.07	4.87	0.963	33.07	6.59	0.931	37.84	1.24	0.960	30.01	5.53	0.956
+ Local regression ($\mathbf{r}^3_b = \boldsymbol{\beta}_b \odot \mathbf{d}$)	$3C$	32.27	4.65	0.963	33.11	6.53	0.948	38.04	1.23	0.967	30.15	5.50	0.960
+ Frequency bands ($\mathbf{r}^{i,j}_b = F_j(\mathbf{r}^i_b)$)	$9C$	32.28	4.65	0.963	33.10	6.56	0.944	38.31	1.21	0.965	30.12	5.49	0.959

Table 5. **Ablation of PanBase integration.** With $\mathbf{s}_{\text{pan}}$, we enable skip replacement, input concatenation, or both under matched training. Original retains the interpolated-LRMS skip and original inputs.

Backbone	Dataset	RR PSNR↑ (dB)				FR HQNR↑			
		Original	Skip only	Input only	**Both (PanBase)**	Original	Skip only	Input only	**Both (PanBase)**
PanNet [45]	WV3	34.29	32.56 −1.73	34.91 +0.62	34.96 +0.67	0.934	0.963 +0.029	0.962 +0.028	0.967 +0.034
	QB	33.83	33.88 +0.05	33.87 +0.04	34.20 +0.37	0.843	0.915 +0.072	0.861 +0.018	0.857 +0.015
	GF2	36.87	38.11 +1.24	37.78 +0.91	37.98 +1.10	0.915	0.976 +0.061	0.929 +0.014	0.927 +0.012
LAGConv [18]	WV3	37.22	37.30 +0.08	37.37 +0.15	37.34 +0.13	0.944	0.963 +0.020	0.963 +0.019	0.958 +0.014
	QB	37.96	38.04 +0.08	37.97 +0.01	38.11 +0.15	0.923	0.928 +0.005	0.846 −0.077	0.911 −0.012
	GF2	42.47	42.87 +0.41	42.81 +0.35	42.82 +0.35	0.915	0.942 +0.028	0.939 +0.025	0.949 +0.034
PINO [22]	WV3	36.42	37.30 +0.88	36.29 −0.13	37.26 +0.84	0.959	0.962 +0.003	0.957 −0.001	0.961 +0.002
	QB	37.37	37.21 −0.15	37.48 +0.11	38.12 +0.75	0.925	0.936 +0.011	0.930 +0.004	0.935 +0.009
	GF2	42.48	43.71 +1.23	42.87 +0.40	44.20 +1.72	0.979	0.981 +0.002	0.979 0.000	0.986 +0.007
Mean change		–	+0.23	+0.27	+0.68	–	+0.026	+0.003	+0.013

(D_λ, D_s) are reported in the *Appendix*.

4.3. Comparison with Existing Backbones

The ten backbones in Table 1 cover CNNs (PanNet [45], LAGConv [18]), mutual-information fusion (MutInf [47]), spatial–frequency fusion (HFIN [30]), attention (PAN-Crafter [10]), binary reconstruction with a diffusion prior (BNNPan [17]), deep unfolding (MDCUN [43]), conditional flows (PanFlowNet [44]), neural operators (PINO [22]), and two-step mixture-of-experts (TS-Pan [21]). Results for other backbones, including state-space models and additional diffusion networks, are reported in the *Appendix*.

Reduced- and full-resolution reconstruction. PanBase improves RR PSNR in 39 of 40 backbone–sensor pairs, by up to 6.93 dB, and HQNR in most settings (Table 1). These gains support supplying PAN structure before residual learning. The geometric mean convergence speed-up is 4.4×, measured by iterations to match each original backbone's best score. Fig. 4 shows clearer roof boundaries and building outlines at both RR and FR.

Cross-sensor zero-shot transfer. As shown in Table 1, RR PSNR improves for every reported backbone on WV2 using WV3-trained models and coefficients without target-sensor recalibration or fine-tuning. Standalone MS-aligned PanBase reaches 30.34 dB, the best PSNR in Table 2, with

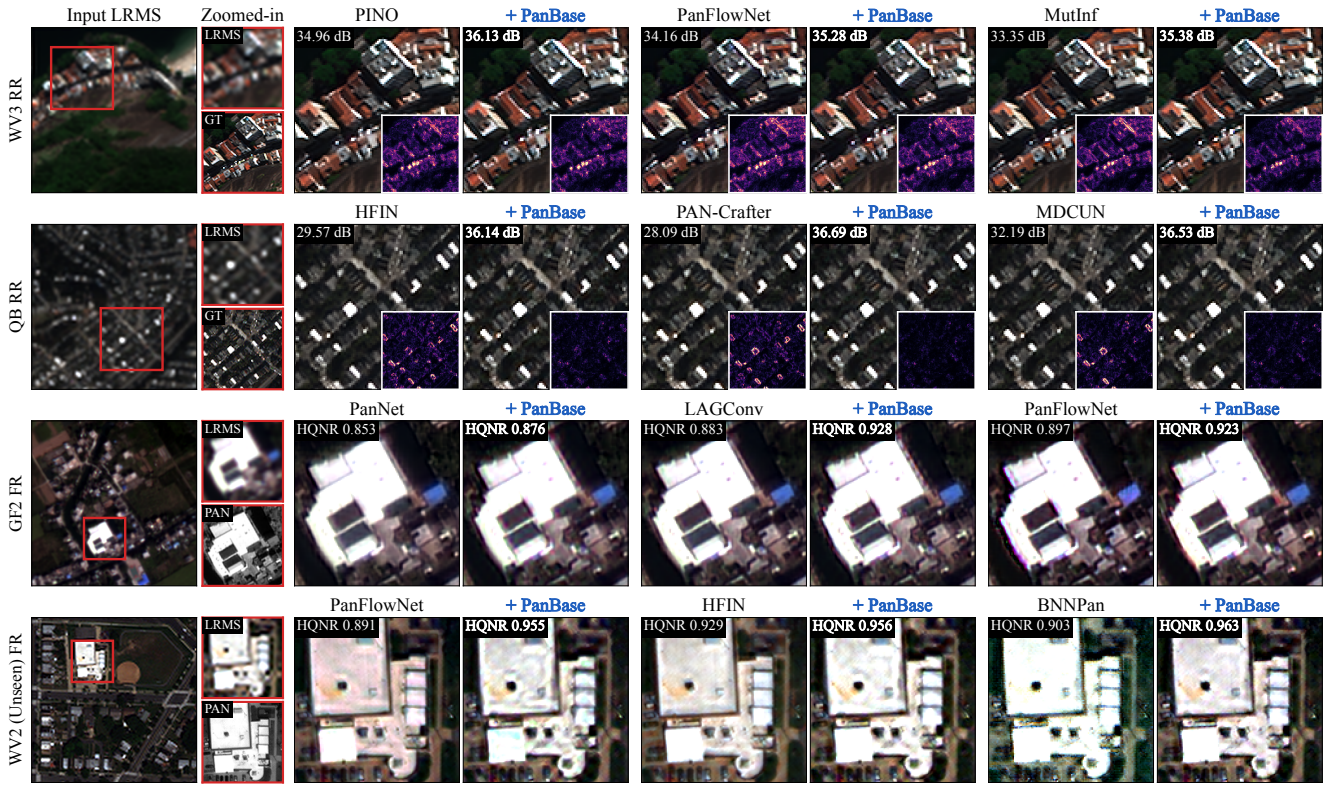

Figure 4. **Qualitative results before and after PanBase integration.** Each pair compares an original backbone with its PanBase-augmented version. WV3/QB rows show RR results with ground-truth references, PSNR, and error insets; GF2/WV2 rows show FR results with PAN references and HQNR. Red boxes mark enlarged regions. Unseen WV2 uses WV3-trained models and coefficients.

competitive quality without a neural network. FR gains are less uniform, while Fig. 4 shows clearer boundaries in the unseen-WV2 examples.

Standalone PanBase versus classical fusion. Table 2 compares standalone PanBase with CS, MRA, and VO methods without neural networks. The PAN-aligned $\mathbf{s}_{\mathrm{pan}}$ improves PSNR, SAM, and HQNR over these methods on all four sensors. Fig. 3 shows clearer boundaries and road structures. The MS-aligned $\mathbf{s}_{\mathrm{ms}}$ further improves RR metrics but lowers HQNR on most sensors; its plug-in results are in the *Appendix*.

4.4. Base Swap Analysis

To isolate the effect of the base, we swap in projected LRMS or classical fused bases under identical integration and training (Table 3). PanBase gives the highest HQNR throughout and the highest PSNR for PanNet. Projected LRMS remains competitive for LAGConv at RR, whereas classical bases can degrade performance.

4.5. Ablation Studies

PanBase construction. Table 4 isolates cumulative contributions. Projection improves RR reconstruction; additive PAN detail gives the largest incremental gain. Multiplicative and local-regression forms further reduce SAM. Frequency decomposition mainly benefits GF2 at RR, with small FR trade-offs. Ablations of the MS-aligned variant $\mathbf{s}_{\mathrm{ms}}$ are reported in the *Appendix*.

PanBase integration. Table 5 separates skip replacement and input concatenation. Combining both gives the largest mean RR gain (0.68 dB) and improves PSNR throughout, supporting exposure of the base to its correction branch. Skip-only integration gives the largest mean FR HQNR gain, while all three integration strategies improve HQNR on average.

5. Conclusion

Why start with a blurry base when sharp PAN detail is already available? Our **PanBase** incorporates this detail into an LRMS-consistent starting point using nine calibrated gains per band, then supplies it through the global skip and backbone input. Across ten backbones and four sensors, it improves RR PSNR in 39 of 40 matched comparisons and supports faster convergence. Reusing WV3 gains on unseen WV2 also yields competitive standalone reconstruction without target-sensor fitting. These findings highlight base construction as an effective choice for residual PS.

Table 6. **Overview of the *Appendix*.**

Section	Contents
Sec. A	Backbone results and training convergence
Sec. B	Standalone fusion and transfer to WV2
Sec. C	MS-aligned base and integration ablations
Sec. D	Sensor model, alignment, and projection
Sec. E	Classical methods and backbone families
Sec. F	Reduced- and full-resolution metrics
Sec. G	Limitations of the plug-in and standalone base

We evaluate how the choice of residual base affects reconstruction quality and training convergence across 15 pansharpening backbones. We also compare the PAN- and MS-aligned bases before neural refinement, including transfer from WV3 to unseen WV2. Table 6 lists the experiments, construction details, evaluation protocol, and limitations.

A. PanBase Across Backbones

Extension to 15 backbones. Table 8 extends the ten-backbone comparison in Table 1 to five additional architectures. With the same training recipe as each original backbone, the PAN-aligned base $\mathbf{s}_{\mathrm{pan}}$ improves reduced-resolution (RR) PSNR in 17 of the 20 additional backbone–sensor pairs and in 56 of 60 pairs overall.

PAN versus MS geometry. Table 9 compares the two geometries on the 53 pairs with results for both. The MS-aligned base $\mathbf{s}_{\mathrm{ms}}$ gives larger mean PSNR gains on each training sensor. The largest difference occurs on GF2, where the mean gain rises from 1.82 to 2.26 dB. On unseen WV2 the PAN- and MS-aligned bases gain 0.54 and 0.36 dB, respectively. For WV2 we reuse the WV3-trained backbone and the WV3 calibration coefficients and construct the base with the WV2 sensor operator.

Training convergence. Fig. 6 plots validation PSNR for the ten backbones. For each training sensor, the target is the best validation PSNR of the original run. We divide the iteration at which the original run first reaches the target by the corresponding iteration for the PanBase run, take the median over reached sensor targets for each backbone, and report the geometric mean across backbones. Across the 14 backbones with convergence results for both geometries, the geometric means are $4.0\times$ for $\mathbf{s}_{\mathrm{pan}}$ and $4.5\times$ for $\mathbf{s}_{\mathrm{ms}}$.

A.1. Results by Sensor

Tables 10, 11, 12, and 13 report all eight RR and three FR metrics. On GF2, all 15 backbones gain PSNR with $\mathbf{s}_{\mathrm{pan}}$, as do all 12 with $\mathbf{s}_{\mathrm{ms}}$ results. On WV2, 14 of 15 improve with $\mathbf{s}_{\mathrm{pan}}$. We also include the two standalone bases in the WV2 table. Without a learned residual, $\mathbf{s}_{\mathrm{ms}}$ reaches 30.34 dB and obtains the highest SSIM (0.8747) among the entries shown. Figs. 7, 8, 9, and 10 pair each reconstruction with its absolute error against HRMS.

B. Standalone PanBase and Cross-Sensor Transfer

Fig. 11 isolates the choice of residual base before any backbone is applied. Interpolated LRMS preserves coarse color but smooths the bright roof boundaries in the first GF2 scene. Injecting PAN detail raises the PSNR of that base from 31.9 to 40.6 dB.

Tables 14, 15, 16, and 17 compare the standalone bases with 12 classical methods. The MS-aligned base has the highest RR PSNR and lowest spectral angle on all four sensors. The PAN-aligned base has higher FR HQNR than every classical method on all four. On unseen WV2, PAN- and MS-aligned PanBase reach 30.12 and 30.34 dB, respectively. Figs. 12 and 13 show the variation across scenes.

C. Ablations in MS Geometry

Construction. Table 18 adds the detail components cumulatively and refits the coefficients at each step. Adding the first, additive PAN detail term to projected LRMS gives the largest PSNR increase on every sensor. On QB this step raises PSNR from 29.09 to 35.30 dB. The multiplicative and regression terms, followed by frequency-dependent weighting, bring it to 36.05 dB. The last step has its largest effect on GF2, increasing PSNR from 39.59 to 39.86 dB.

Backbone integration. Table 19 separates the two ways the backbone uses the base. Across PanNet and PINO on three training sensors, changing only the global skip gains 1.43 dB on average. Supplying the base only as an input gains 1.25 dB, and using it in both places gains 1.78 dB. Input concatenation alone gives the largest mean HQNR gain (0.029), followed by using both (0.023). Table 20 holds the integration fixed and varies the base. For PanNet on GF2, projected LRMS yields 37.62 dB and 0.915 HQNR, while the complete MS-aligned base yields 40.06 dB and 0.963.

D. PanBase Construction Details

Sensor model. For each MS band b, we construct a 41×41 MTF-matched kernel $\mathbf{h}_b$ from a Gaussian frequency response with gain G_b at the MS Nyquist frequency, using a Kaiser window ($\beta = 0.5$) [2]. We use the PanCollection gains [9] for WV3 $(0.325, 0.355, 0.360, 0.350, 0.365, 0.360, 0.335, 0.315)$, QB $(0.34, 0.32, 0.30, 0.22)$, and WV2 (0.35 for the first seven bands and 0.27 for NIR2). For GF2, we set $G_b = 0.30$ for every band. We use the same filter design for the PAN smoothing kernel $\bar{\mathbf{h}}$, with the mean gain $\bar{G} = C^{-1} \sum_b G_b$. The degradation operator $\mathbf{D}_b$ applies $\mathbf{h}_b$ with replicate padding and samples every fourth pixel with a two-pixel offset along each axis.

Frequency decomposition. Let $\nu = 2r\|\mathbf{f}\|_2$, where $r = 4$ and $\mathbf{f}$ is measured in cycles per PAN pixel. With $a_k(\nu) = \sigma((\nu - k)/0.15)$ and logistic function σ, the low-, middle-, and high-frequency weights are

$$(\alpha_1, \alpha_2, \alpha_3) = (1 - a_1,\ a_1 - a_2,\ a_2). \tag{13}$$

We apply these weights to each injection component after forming the pointwise products, using reflection padding of $\max(4, \lfloor \min(H, W)/4 \rfloor)$ pixels before the FFT and cropping after the inverse FFT (Fig. 5).

Injection maps and MS alignment. In Eq. (2), we floor the denominator of the intensity ratio at $10^{-3}\,\mathrm{mean}(|\mathbf{x}^{\mathrm{lr}}_{\mathrm{pan}}|)$. For local regression, we use a replicate-padded 7×7 window on the LRMS grid, floor the local PAN variance at $10^{-6}\,\mathrm{var}(\mathbf{p})$, clip the slope to $[-2, 4]$, and upsample it bilinearly. For $\mathbf{s}_{\mathrm{ms}}$, we fit $\sum_b w_b \mathbf{x}^{\mathrm{lr}}_{\mathrm{ms},b} + c$ to the low-pass PAN image by least squares, using the 23-tap LRMS interpolation supplied by PanCollection. We estimate one translation per image by Hann-windowed phase correlation over $\nu < 1.5$, with translation magnitude bounded by $\min(32, \min(H, W)/4)$ pixels. We refine the shift by parabolic interpolation and up to ten Gauss–Newton steps, stopping when the update is shorter than 10^{-3} pixels. We apply a Fourier phase ramp on a reflect-extended grid to the PAN detail and the PAN terms in $\mathbf{m}_b$ and $\boldsymbol{\beta}_b$.

Projection. We solve each bandwise system involving $\mathbf{D}_b\mathbf{D}_b^\top$ by conjugate gradients with the exact adjoint $\mathbf{D}_b^\top$. The preconditioner uses a separable approximation to $\mathbf{h}_b$ formed from its marginals. We stop when the squared residual falls below 10^{-12} of its initial value or after 200 iterations. Before clipping, the relative LRMS residual $\|\mathbf{D}_b\mathbf{s}_b - \mathbf{x}_{\mathrm{ms},b}\|_2 / \|\mathbf{x}_{\mathrm{ms},b}\|_2$ is approximately 4×10^{-7} in float32. We clip the base to $[0, 2047]$ for WV3, QB, and WV2 and to $[0, 1023]$ for GF2. We fit the $9C$ coefficients once per training sensor on 6,000 training patches. Both geometries use their respective fixed coefficients throughout backbone training and testing.

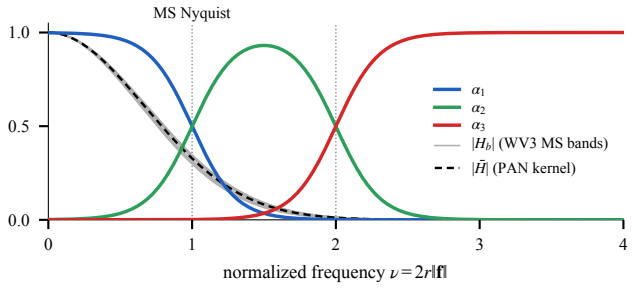

Figure 5. **Frequency weights for PAN detail injection.** The three weights overlap near $\nu = 1$ and 2 and sum to one. Gray curves show the WV3 MS filter responses and the dashed curve shows the PAN smoothing filter. Along the plotted axis, $\nu = 1$ and 4 are the MS and PAN Nyquist frequencies.

E. Comparison Methods

We compare standalone PanBase with three families of classical fusion methods [32, 36]. Component substitution transfers PAN detail through an MS-derived intensity component. Multiresolution analysis extracts PAN detail across scales and injects it into the MS bands. Variational optimization reconstructs the HRMS image from the observations and a regularization term. Interpolated LRMS provides a common reference for these comparisons.

Table 7 groups the 15 learned backbones by the network used for reconstruction. BNNPan uses a binary convolutional backbone with an auxiliary diffusion prior. For the four backbones that sample at inference (PanFlowNet, BNNPan, U-Know-DiffPAN, and PanDiT), we report one sample per test image with a fixed random seed. U-Know-DiffPAN and PanDiT use 25 DDIM steps in our evaluation.

F. Evaluation Metrics

We compute each metric per image and average over the 20 test images of each sensor and resolution, following PanCollection [9].

Reduced resolution. We compare the fused image with its HRMS reference on the same 197×197 interior crop of each 256×256 image. We divide digital numbers by 2,047 on WV3, QB, and WV2 and by 1,023 on GF2. PSNR uses a peak value of one and the mean squared error over all pixels and bands. SSIM [39] is averaged across bands using an 11×11 Gaussian window ($\sigma = 1.5$). SAM is the mean pixelwise spectral angle in degrees. ERGAS is $(100/r)\sqrt{C^{-1}\sum_{b=1}^{C}(\mathrm{RMSE}_b/\mu_b)^2}$, where μ_b is the mean of reference band b and $r = 4$. SCC averages the bandwise correlations between the fused and reference images after a 3×3 Laplacian filter. We compute $Q2^n$ [13] on the original digital numbers in non-overlapping 32×32 blocks, using Q_4 for four bands and Q_8 for eight.

For LPIPS and DISTS, we form RGB composites from bands 5, 3, and 2 on WV3 and WV2, and bands 3, 2, and 1 on QB and GF2 (one-based indexing). The 1st and 99th percentiles of each reference band define a linear stretch shared by the reference and all reconstructions of that image. We use the `pyiqa` implementation for both distances.

Full resolution. We evaluate the entire 512×512 fused image with the benchmark toolbox [35, 36]. Spectral distortion D_λ is one minus $Q2^n$ between the observed LRMS and the fused image after MS MTF filtering and $4\times$ decimation. For this evaluation we use the QB and WV3 MTF presets on their respective sensors, the WV3 preset on WV2, and a Nyquist gain of 0.3 on GF2. Spatial distortion D_s is the mean across bands of the absolute difference between the local MS–PAN quality indices at the fused and degraded scales [4]. We retain the toolbox's 33×33 sliding window and SSIM data-range convention for digital numbers. The

Table 7. **Backbones evaluated with PanBase.** Methods are grouped by their reconstruction architecture.

Family	Backbone	Main components
CNN	PanNet [45]	High-pass residual network with an upsampled-MS skip
	LAGConv [18]	Residual network with locally adaptive convolutions
	MutInf [47]	Edge-aware PAN and MS features fused by invertible coupling blocks
	MDCUN [43]	Six-stage deep unfolding with memory and convolutional priors
	PanFlowNet [44]	Conditional normalizing flow with convolutional coupling blocks
	MSDDN [15]	Invertible blocks for multiscale spatial and Fourier-domain fusion
	HFIN [30]	Hierarchical fusion of spatial, Fourier, and short-time Fourier features
	UTeRM [24]	Ten-stage tensor-rank minimization with convolutional proximal operators
	BNNPan [17]	Binary wavelet CNN with a latent diffusion prior
	PAN-Crafter [10]	Convolutional U-Net with local attention for PAN–MS alignment
Transformer	PINO [22]	Window and channel attention encoder with a Galerkin-attention decoder
	TS-Pan [21]	Convolutional experts followed by attention and RWKV refinement
State-space	MMMamba [38]	Selective scans over interleaved MS and PAN windows
Diffusion	U-Know-DiffPAN [20]	U-Net student distilled from a diffusion teacher with uncertainty weighting
	PanDiT [12]	Diffusion Transformer with DDIM sampling

per-image score is $\mathrm{HQNR} = (1 - D_\lambda)(1 - D_s)$.

G. Limitations

Some recent backbones gain little from replacing the residual base. On WV3, $\mathbf{s}_{\mathrm{pan}}$ improves PAN-Crafter by only 0.06 dB and reduces TS-Pan PSNR by 0.40 dB. On GF2, both still improve, by 0.48 and 2.40 dB, respectively.

Standalone $\mathbf{s}_{\mathrm{ms}}$ has lower FR HQNR than $\mathbf{s}_{\mathrm{pan}}$ on WV3, QB, and WV2. The gap is largest on QB. MS alignment raises RR PSNR from 33.10 to 36.05 dB, but spatial distortion D_s rises from 0.0525 to 0.2953 and HQNR falls from 0.944 to 0.703.

Table 8. **PanBase in PAN geometry on five additional backbones.** Highlighted rows use $\mathbf{s}_{\mathrm{pan}}$ as the global residual base and an additional input under matched training. Deltas are relative to each original backbone. **Bold** and underline mark the best and second-best scores per column. WV2† uses WV3-trained models without fine-tuning. For each backbone, speed-up is the median ratio of iterations to reach the original best score across reached WV3, QB, and GF2 targets. We report the geometric mean across backbones.

Method	Venue	RR PSNR↑ (dB)				FR HQNR↑				Training speed-up↑
		WV3	QB	GF2	WV2†	WV3	QB	GF2	WV2†	
MSDDN [15]	TGRS'23	35.62	36.47	38.72	28.65	0.956	0.910	0.945	0.931	–
+ PanBase ($\mathbf{s}_{\mathrm{pan}}$)	–	36.66 +1.04	**38.11** +1.63	41.80 +3.08	28.82 +0.17	0.928 −0.028	0.941 +0.031	0.956 +0.011	0.920 −0.011	6.7×
UTeRM [24]	TGRS'24	35.96	37.26	41.89	28.85	0.942	0.949	0.940	0.954	–
+ PanBase ($\mathbf{s}_{\mathrm{pan}}$)	–	36.45 +0.50	37.40 +0.14	42.56 +0.67	29.13 +0.28	0.956 +0.014	0.941 −0.007	0.966 +0.026	0.922 −0.032	1.5×
U-Know-DiffPAN [20]	CVPR'25	37.44	37.00	43.24	29.20	0.964	0.939	0.966	0.966	–
+ PanBase ($\mathbf{s}_{\mathrm{pan}}$)	–	**37.83** +0.39	37.45 +0.45	**44.05** +0.81	29.80 +0.60	0.959 −0.004	0.818 −0.121	0.976 +0.010	0.947 −0.019	3.0×
PanDiT [12]	TGRS'25	36.26	37.10	42.00	29.15	0.964	0.944	0.954	**0.978**	–
+ PanBase ($\mathbf{s}_{\mathrm{pan}}$)	–	36.13 −0.13	36.67 −0.43	42.16 +0.16	**30.17** +1.03	**0.965** +0.001	**0.953** +0.009	**0.982** +0.027	0.962 −0.016	1.4×
MMMamba [38]	AAAI'26	36.23	36.97	39.68	28.89	0.952	0.788	0.928	0.932	–
+ PanBase ($\mathbf{s}_{\mathrm{pan}}$)	–	37.05 +0.81	37.75 +0.78	42.72 +3.04	28.88 −0.02	0.952 0.000	0.916 +0.127	0.961 +0.034	0.899 −0.033	2.4×
Mean change ($\mathbf{s}_{\mathrm{pan}}$)		+0.52	+0.51	+1.55	+0.41	−0.003	+0.008	+0.022	−0.022	2.5×

Table 9. **PanBase in MS geometry across 15 backbones.** Highlighted rows use $\mathbf{s}_{\mathrm{ms}}$ as the global residual base and an additional input under matched training. Deltas are relative to each original backbone. **Bold** and underline mark the best and second-best scores per column. WV2† uses WV3-trained models without fine-tuning. For each backbone, speed-up is the median ratio of iterations to reach the original best score across reached WV3, QB, and GF2 targets. We report the geometric mean across backbones. Dashes indicate unavailable results. The final row uses the same available backbone–dataset pairs for $\mathbf{s}_{\mathrm{pan}}$.

Method	Venue	RR PSNR↑ (dB)				FR HQNR↑				Training speed-up↑
		WV3	QB	GF2	WV2†	WV3	QB	GF2	WV2†	
PanNet [45]	ICCV'17	34.29	33.83	36.87	28.51	0.934	0.843	0.915	0.968	–
+ PanBase ($\mathbf{s}_{\mathrm{ms}}$)	–	35.22 +0.93	36.41 +2.57	40.06 +3.18	29.40 +0.89	**0.967** +0.034	0.881 +0.038	0.963 +0.047	0.963 −0.005	60.0×
LAGConv [18]	AAAI'22	37.22	37.96	42.47	28.71	0.944	0.923	0.915	0.949	–
+ PanBase ($\mathbf{s}_{\mathrm{ms}}$)	–	37.29 +0.07	38.12 +0.16	43.02 +0.55	29.07 +0.36	0.962 +0.018	0.900 −0.022	0.969 +0.055	0.936 −0.013	1.7×
MutInf [47]	CVPR'22	35.77	36.64	36.65	28.87	0.913	0.876	0.943	0.902	–
+ PanBase ($\mathbf{s}_{\mathrm{ms}}$)	–	36.97 +1.20	38.29 +1.65	43.03 +6.38	29.23 +0.36	0.964 +0.052	0.927 +0.051	0.984 +0.041	0.935 +0.033	12.3×
MDCUN [43]	CVPR'22	36.95	35.99	42.04	28.96	0.961	0.914	0.953	0.938	–
+ PanBase ($\mathbf{s}_{\mathrm{ms}}$)	–	36.92 −0.03	38.08 +2.09	43.15 +1.11	28.79 −0.18	0.961 0.000	0.937 +0.023	**0.985** +0.032	0.941 +0.003	9.4×
PanFlowNet [44]	ICCV'23	35.76	36.23	38.56	27.78	0.918	0.853	0.904	0.869	–
+ PanBase ($\mathbf{s}_{\mathrm{ms}}$)	–	36.56 +0.80	37.93 +1.70	40.19 +1.62	28.22 +0.44	0.865 −0.053	0.909 +0.055	0.934 +0.030	0.912 +0.043	7.5×
MSDDN [15]	TGRS'23	35.62	36.47	38.72	28.65	0.956	0.910	0.945	0.931	–
+ PanBase ($\mathbf{s}_{\mathrm{ms}}$)	–	36.85 +1.24	38.08 +1.61	41.86 +3.14	28.68 +0.03	0.942 −0.014	0.946 +0.035	0.971 +0.027	0.899 −0.031	7.1×
HFIN [30]	CVPR'24	36.53	35.17	38.08	28.39	0.917	0.796	0.909	0.903	–
+ PanBase ($\mathbf{s}_{\mathrm{ms}}$)	–	37.40 +0.87	38.03 +2.86	42.56 +4.48	28.68 +0.29	0.966 +0.050	0.896 +0.101	0.966 +0.057	0.922 +0.019	12.7×
UTeRM [24]	TGRS'24	35.96	37.26	41.89	28.85	0.942	**0.949**	0.940	0.954	–
+ PanBase ($\mathbf{s}_{\mathrm{ms}}$)	–	36.46 +0.50	37.54 +0.28	42.69 +0.80	29.45 +0.60	0.963 +0.021	0.894 −0.055	0.974 +0.034	0.946 −0.009	1.4×
BNNPan [17]	CVPR'25	36.73	37.38	42.12	27.35	0.936	0.856	0.927	0.755	–
+ PanBase ($\mathbf{s}_{\mathrm{ms}}$)	–	–	–	–	–	–	–	–	–	–
U-Know-DiffPAN [20]	CVPR'25	37.44	37.00	43.24	29.20	0.964	0.939	0.966	0.966	–
+ PanBase ($\mathbf{s}_{\mathrm{ms}}$)	–	37.89 +0.45	37.90 +0.90	–	30.04 +0.84	0.963 0.000	0.871 −0.068	–	0.960 −0.006	2.2×
PAN-Crafter [10]	ICCV'25	37.48	35.75	44.33	29.04	0.964	0.826	0.971	0.949	–
+ PanBase ($\mathbf{s}_{\mathrm{ms}}$)	–	37.60 +0.12	37.93 +2.18	**45.00** +0.67	28.88 −0.16	0.960 −0.004	0.935 +0.109	0.981 +0.010	0.933 −0.015	1.7×
PINO [22]	NeurIPS'25	36.42	37.37	42.48	29.76	0.959	0.925	0.979	0.946	–
+ PanBase ($\mathbf{s}_{\mathrm{ms}}$)	–	37.55 +1.13	38.03 +0.66	44.66 +2.19	**30.43** +0.67	0.954 −0.005	0.945 +0.020	0.984 +0.004	0.941 −0.005	12.0×
PanDiT [12]	TGRS'25	36.26	37.10	42.00	29.15	0.964	0.944	0.954	**0.978**	–
+ PanBase ($\mathbf{s}_{\mathrm{ms}}$)	–	36.47 +0.21	37.40 +0.30	42.34 +0.35	30.37 +1.22	0.962 −0.002	0.837 −0.107	0.978 +0.024	0.962 −0.016	1.3×
MMMamba [38]	AAAI'26	36.23	36.97	39.68	28.89	0.952	0.788	0.928	0.932	–
+ PanBase ($\mathbf{s}_{\mathrm{ms}}$)	–	36.70 +0.46	–	–	27.96 −0.93	0.954 +0.002	–	–	0.877 −0.056	2.0×
TS-Pan [21]	TGRS'26	**37.90**	**38.53**	41.94	29.59	0.946	0.743	0.944	0.920	–
+ PanBase ($\mathbf{s}_{\mathrm{ms}}$)	–	37.83 −0.08	38.49 −0.04	44.56 +2.62	30.15 +0.56	0.934 −0.012	0.868 +0.124	0.979 +0.034	0.935 +0.015	1.2×
Mean change ($\mathbf{s}_{\mathrm{ms}}$)		+0.56	+1.30	+2.26	+0.36	+0.006	+0.023	+0.033	−0.003	4.5×
Mean change ($\mathbf{s}_{\mathrm{pan}}$), same cells		+0.49	+0.95	+1.82	+0.54	+0.013	+0.024	+0.016	+0.001	4.0×

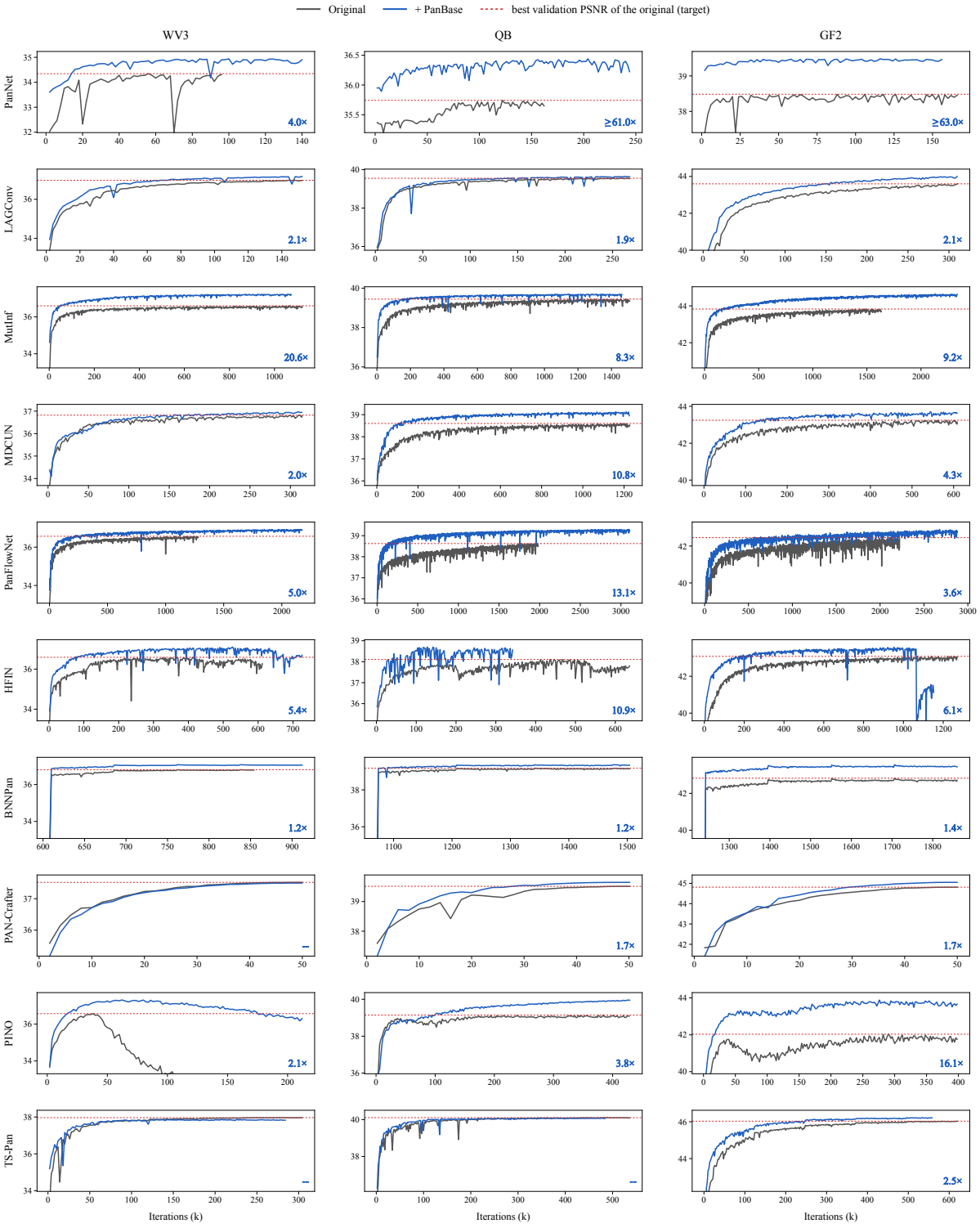

Figure 6. **Validation PSNR over training iterations.** Each panel shows one backbone trained with its original base and with PanBase. The dotted horizontal line marks the best validation PSNR of the original run. Annotations give the iteration ratio at that target. A dash marks an unreached target. For PanNet, the QB and GF2 panels show lower bounds. We use $60\times$ for PanNet when computing the tabulated geometric means.

Table 10. **PanBase results on WV3.** Highlighted rows use PanBase as the residual base and an additional input. Green and red indicate gains and drops relative to each original backbone. **Bold** and underline mark the best and second-best scores per metric. Means exclude missing entries (–).

Method	Reduced resolution (HRMS reference)								Full resolution (no reference)		
	PSNR↑	SSIM↑	SAM↓	ERGAS↓	SCC↑	$Q2^n$↑	LPIPS↓	DISTS↓	D_λ↓	D_s↓	HQNR↑
PanNet [45]	34.29	0.9467	4.24	3.14	0.8046	0.8719	0.0808	0.1187	0.0107	0.0564	0.934
+ PanBase (s_{pan})	34.96	0.9531	3.89	2.92	0.8132	0.8885	0.0510	0.0895	0.0032	0.0296	0.967
+ PanBase (s_{ms})	35.22	0.9546	3.91	2.84	0.8252	0.8812	0.0574	0.0900	0.0047	0.0281	0.967
LAGConv [18]	37.22	0.9713	3.09	2.25	0.8739	0.9099	0.0317	0.0622	0.0132	0.0438	0.944
+ PanBase (s_{pan})	37.34	0.9719	3.05	2.21	0.8753	0.9141	0.0275	0.0599	0.0056	0.0368	0.958
+ PanBase (s_{ms})	37.29	0.9716	3.08	2.23	0.8732	0.9120	0.0289	0.0602	0.0061	0.0323	0.962
MutInf [47]	35.77	0.9641	3.55	2.72	0.8408	0.9030	0.0386	0.0679	0.0320	0.0577	0.913
+ PanBase (s_{pan})	37.05	0.9712	3.10	2.29	0.8693	0.9143	0.0286	0.0578	0.0040	0.0290	0.967
+ PanBase (s_{ms})	36.97	0.9709	3.15	2.32	0.8670	0.9153	0.0291	0.0563	0.0063	0.0297	0.964
MDCUN [43]	36.95	0.9709	3.08	2.31	0.8680	0.9156	0.0301	0.0622	0.0080	0.0314	0.961
+ PanBase (s_{pan})	37.12	0.9710	3.06	2.27	0.8701	0.9171	0.0302	0.0601	0.0038	0.0290	0.967
+ PanBase (s_{ms})	36.92	0.9702	3.12	2.33	0.8648	0.9161	0.0286	0.0590	0.0034	0.0356	0.961
PanFlowNet [44]	35.76	0.9657	3.61	2.63	0.8512	0.9034	0.0376	0.0736	0.0276	0.0565	0.918
+ PanBase (s_{pan})	36.53	0.9679	3.33	2.46	0.8547	0.9039	0.0335	0.0626	0.0116	0.0346	0.954
+ PanBase (s_{ms})	36.56	0.9686	3.27	2.44	0.8602	0.9045	0.0328	0.0630	0.0501	0.0997	0.865
MSDDN [15]	35.62	0.9655	3.49	2.68	0.8582	0.8968	0.0360	0.0705	0.0215	**0.0234**	0.956
+ PanBase (s_{pan})	36.66	0.9692	3.26	2.39	0.8643	0.9091	0.0314	0.0598	0.0069	0.0658	0.928
+ PanBase (s_{ms})	36.85	0.9696	3.22	2.37	0.8705	0.9100	0.0290	0.0583	0.0080	0.0507	0.942
HFIN [30]	36.53	0.9696	3.31	2.46	0.8687	0.8987	0.0361	0.0682	0.0300	0.0556	0.917
+ PanBase (s_{pan})	37.26	0.9720	3.03	2.23	0.8740	0.9166	0.0297	0.0592	0.0050	0.0301	0.965
+ PanBase (s_{ms})	37.40	0.9726	3.01	2.19	0.8766	0.9169	0.0304	0.0590	0.0052	0.0289	0.966
UTeRM [24]	35.96	0.9630	3.39	2.59	0.8436	0.9025	0.0417	0.0762	0.0165	0.0422	0.942
+ PanBase (s_{pan})	36.45	0.9667	3.30	2.45	0.8564	0.9077	0.0340	0.0653	0.0140	0.0303	0.956
+ PanBase (s_{ms})	36.46	0.9664	3.29	2.45	0.8548	0.9079	0.0362	0.0692	0.0085	0.0287	0.963
BNNPan [17]	36.73	0.9691	3.20	2.37	0.8641	0.9110	0.0318	0.0657	0.0170	0.0484	0.936
+ PanBase (s_{pan})	37.05	0.9710	3.10	2.28	0.8686	0.9144	0.0289	0.0589	0.0067	0.0242	**0.969**
+ PanBase (s_{ms})	–	–	–	–	–	–	–	–	–	–	–
U-Know-DiffPAN [20]	37.44	0.9728	2.91	2.18	0.8783	0.9191	0.0297	0.0618	0.0070	0.0298	0.964
+ PanBase (s_{pan})	37.83	0.9747	2.83	2.09	0.8843	0.9224	0.0250	0.0566	0.0044	0.0363	0.959
+ PanBase (s_{ms})	37.89	0.9750	**2.82**	**2.07**	**0.8859**	**0.9226**	0.0244	0.0561	0.0047	0.0321	0.963
PAN-Crafter [10]	37.48	0.9733	2.93	2.18	0.8772	0.9183	0.0284	0.0591	0.0090	0.0269	0.964
+ PanBase (s_{pan})	37.54	0.9738	2.93	2.16	0.8764	0.9206	0.0249	0.0559	0.0039	0.0341	0.962
+ PanBase (s_{ms})	37.60	0.9739	2.91	2.14	0.8784	0.9199	0.0258	0.0575	0.0055	0.0344	0.960
PINO [22]	36.42	0.9675	3.36	2.48	0.8554	0.9095	0.0440	0.0737	0.0101	0.0312	0.959
+ PanBase (s_{pan})	37.26	0.9717	3.07	2.24	0.8686	0.9136	0.0271	0.0570	0.0035	0.0353	0.961
+ PanBase (s_{ms})	37.55	0.9735	2.99	2.17	0.8774	0.9188	0.0239	0.0543	0.0042	0.0423	0.954
PanDiT [12]	36.26	0.9655	3.31	2.51	0.8440	0.8998	0.0343	0.0670	0.0050	0.0308	0.964
+ PanBase (s_{pan})	36.13	0.9641	3.37	2.56	0.8370	0.9018	0.0313	0.0638	**0.0012**	0.0338	0.965
+ PanBase (s_{ms})	36.47	0.9664	3.30	2.45	0.8463	0.9042	0.0294	0.0613	0.0015	0.0366	0.962
MMMamba [38]	36.23	0.9690	3.29	2.55	0.8581	0.9150	0.0269	0.0565	0.0045	0.0436	0.952
+ PanBase (s_{pan})	37.05	0.9720	3.02	2.33	0.8697	0.9188	0.0233	**0.0511**	0.0069	0.0413	0.952
+ PanBase (s_{ms})	36.70	0.9711	3.11	2.34	0.8676	0.9181	0.0265	0.0531	0.0041	0.0424	0.954
TS-Pan [21]	**37.90**	**0.9754**	2.84	**2.07**	0.8847	0.9173	0.0230	0.0543	0.0051	0.0495	0.946
+ PanBase (s_{pan})	37.51	0.9716	3.01	2.19	0.8593	0.9061	0.0326	0.1059	0.0015	0.0432	0.955
+ PanBase (s_{ms})	37.83	0.9750	2.89	2.08	0.8814	0.9125	**0.0226**	0.0550	0.0030	0.0632	0.934
Mean change (s_{pan})	+0.48	+0.0022	−0.15	−0.14	+0.0047	+0.0051	−0.0061	−0.0050	−0.0090	−0.0062	+0.015
Mean change (s_{ms})	+0.56	+0.0028	−0.17	−0.17	+0.0087	+0.0057	−0.0067	−0.0085	−0.0060	+0.0004	+0.006

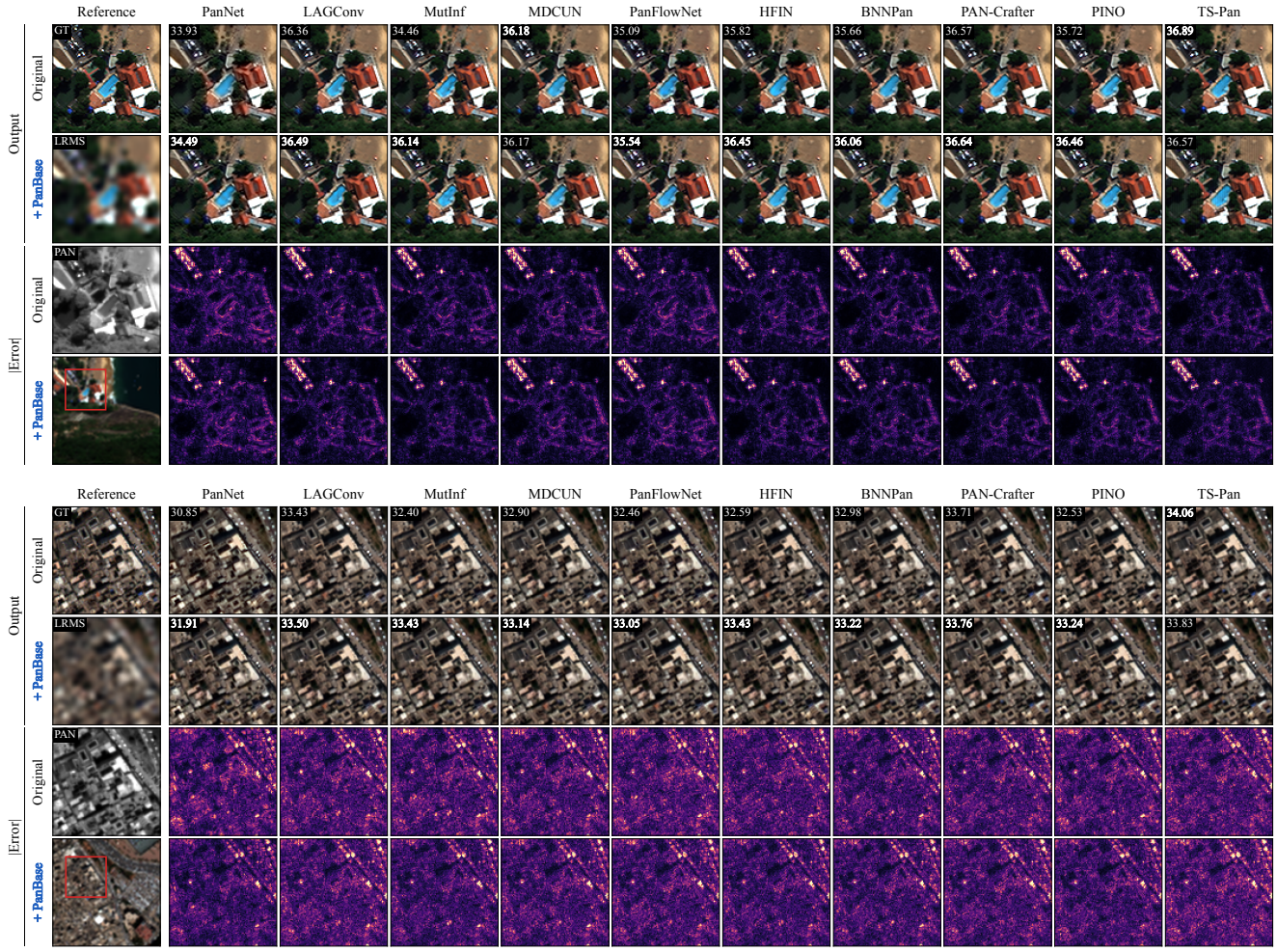

Figure 7. **Backbone reconstructions on WV3 at reduced resolution.** Each column pairs the original and PanBase outputs with their absolute error maps for two scenes. Scores are PSNR in dB. The reference column identifies the crop and shows the HRMS, interpolated LRMS, and PAN images.

Table 11. **PanBase results on QB.** Highlighted rows use PanBase as the residual base and an additional input. Green and red indicate gains and drops relative to each original backbone. **Bold** and underline mark the best and second-best scores per metric. Means exclude missing entries (–).

Method	Reduced resolution (HRMS reference)								Full resolution (no reference)		
	PSNR↑	SSIM↑	SAM↓	ERGAS↓	SCC↑	$Q2^n$↑	LPIPS↓	DISTS↓	D_λ↓	D_s↓	HQNR↑
PanNet [45]	33.83	0.9084	6.35	6.01	0.7354	0.8826	0.1197	0.1565	0.0262	0.1350	0.843
+ PanBase ($\mathbf{s}_{\text{pan}}$)	34.20	0.9125	6.04	5.75	0.7378	0.8889	0.0947	0.1302	0.0409	0.1064	0.857
+ PanBase ($\mathbf{s}_{\text{ms}}$)	36.41	0.9405	5.53	4.45	0.8315	0.9178	0.0691	0.1128	0.0502	0.0723	0.881
LAGConv [18]	37.96	0.9598	4.54	3.73	0.8636	0.9344	0.0583	0.0960	0.0245	0.0539	0.923
+ PanBase ($\mathbf{s}_{\text{pan}}$)	38.11	0.9607	4.50	3.66	0.8658	0.9374	0.0502	0.0916	0.0231	0.0677	0.911
+ PanBase ($\mathbf{s}_{\text{ms}}$)	38.12	0.9609	4.50	3.66	0.8645	0.9369	0.0497	0.0910	0.0230	0.0782	0.900
MutInf [47]	36.64	0.9510	5.07	4.37	0.8416	0.9174	0.0704	0.1095	0.0609	0.0681	0.876
+ PanBase ($\mathbf{s}_{\text{pan}}$)	38.33	0.9628	4.41	3.57	0.8693	0.9393	0.0501	0.0912	0.0392	0.0583	0.905
+ PanBase ($\mathbf{s}_{\text{ms}}$)	38.29	0.9628	4.41	3.59	0.8664	0.9389	0.0560	0.0902	0.0353	0.0394	0.927
MDCUN [43]	35.99	0.9451	4.92	4.73	0.8089	0.9153	0.0663	0.1055	0.0321	0.0564	0.914
+ PanBase ($\mathbf{s}_{\text{pan}}$)	37.66	0.9580	4.56	3.86	0.8553	0.9335	0.0573	0.0961	0.0153	0.0426	0.943
+ PanBase ($\mathbf{s}_{\text{ms}}$)	38.08	0.9606	4.45	3.67	0.8642	0.9364	0.0542	0.0937	0.0213	0.0427	0.937
PanFlowNet [44]	36.23	0.9419	5.14	4.72	0.7913	0.9123	0.0653	0.1053	0.0712	0.0813	0.853
+ PanBase ($\mathbf{s}_{\text{pan}}$)	37.87	0.9589	4.62	3.76	0.8626	0.9329	0.0557	0.0937	0.1184	0.0575	0.831
+ PanBase ($\mathbf{s}_{\text{ms}}$)	37.93	0.9590	4.62	3.74	0.8643	0.9343	0.0518	0.0944	0.0426	0.0509	0.909
MSDDN [15]	36.47	0.9459	5.29	4.43	0.8451	0.9199	0.0572	0.1024	0.0503	0.0415	0.910
+ PanBase ($\mathbf{s}_{\text{pan}}$)	38.11	0.9608	4.55	3.66	0.8672	0.9368	0.0489	**0.0846**	0.0247	0.0353	0.941
+ PanBase ($\mathbf{s}_{\text{ms}}$)	38.08	0.9607	4.52	3.67	0.8651	0.9376	0.0510	0.0899	0.0238	0.0313	0.946
HFIN [30]	35.17	0.9263	5.45	5.35	0.7545	0.8920	0.0819	0.1187	0.1024	0.1141	0.796
+ PanBase ($\mathbf{s}_{\text{pan}}$)	37.42	0.9544	4.77	3.97	0.8524	0.9306	0.0547	0.0939	0.0341	0.0436	0.924
+ PanBase ($\mathbf{s}_{\text{ms}}$)	38.03	0.9599	4.50	3.70	0.8650	0.9355	0.0543	0.0941	0.0402	0.0665	0.896
UTeRM [24]	37.26	0.9544	4.68	4.04	0.8462	0.9278	0.0755	0.1063	0.0197	0.0324	0.949
+ PanBase ($\mathbf{s}_{\text{pan}}$)	37.40	0.9552	4.69	3.97	0.8512	0.9280	0.0731	0.1061	0.0303	**0.0295**	0.941
+ PanBase ($\mathbf{s}_{\text{ms}}$)	37.54	0.9564	4.63	3.90	0.8536	0.9308	0.0670	0.1042	0.0621	0.0469	0.894
BNNPan [17]	37.38	0.9541	4.82	3.99	0.8515	0.9293	0.0571	0.1004	0.0451	0.1032	0.856
+ PanBase ($\mathbf{s}_{\text{pan}}$)	37.71	0.9570	4.70	3.83	0.8571	0.9326	0.0534	0.0945	0.0307	0.0923	0.880
+ PanBase ($\mathbf{s}_{\text{ms}}$)	–	–	–	–	–	–	–	–	–	–	–
U-Know-DiffPAN [20]	37.00	0.9534	4.61	4.15	0.8188	0.9304	0.0570	0.0955	0.0211	0.0411	0.939
+ PanBase ($\mathbf{s}_{\text{pan}}$)	37.45	0.9572	4.48	3.94	0.8305	0.9330	0.0567	0.0916	0.1018	0.0893	0.818
+ PanBase ($\mathbf{s}_{\text{ms}}$)	37.90	0.9602	4.38	3.75	0.8468	0.9368	0.0526	0.0897	0.0536	0.0798	0.871
PAN-Crafter [10]	35.75	0.9271	5.34	5.23	0.7378	0.8920	0.0853	0.1097	0.0684	0.1133	0.826
+ PanBase ($\mathbf{s}_{\text{pan}}$)	37.74	0.9571	4.74	3.83	0.8553	0.9336	0.0563	0.0972	0.0474	0.1116	0.846
+ PanBase ($\mathbf{s}_{\text{ms}}$)	37.93	0.9605	4.53	3.73	0.8580	0.9368	0.0480	0.0923	0.0180	0.0479	0.935
PINO [22]	37.37	0.9529	5.40	4.00	0.8509	0.9258	0.0947	0.1279	0.0200	0.0558	0.925
+ PanBase ($\mathbf{s}_{\text{pan}}$)	38.12	0.9604	4.59	3.65	0.8648	0.9365	0.0577	0.0920	0.0231	0.0431	0.935
+ PanBase ($\mathbf{s}_{\text{ms}}$)	38.03	0.9602	4.53	3.69	0.8616	0.9375	0.0532	0.0911	0.0176	0.0376	0.945
PanDiT [12]	37.10	0.9522	4.91	4.11	0.8406	0.9282	0.0619	0.0993	0.0146	0.0424	0.944
+ PanBase ($\mathbf{s}_{\text{pan}}$)	36.67	0.9481	4.98	4.32	0.8265	0.9244	0.0603	0.0949	0.0059	0.0417	**0.953**
+ PanBase ($\mathbf{s}_{\text{ms}}$)	37.40	0.9541	4.81	3.97	0.8473	0.9296	0.0573	0.0917	**0.0049**	0.1590	0.837
MMMamba [38]	36.97	0.9561	4.74	4.19	0.8294	0.9269	0.0505	0.0929	0.0633	0.1586	0.788
+ PanBase ($\mathbf{s}_{\text{pan}}$)	37.75	0.9605	4.46	3.81	0.8514	0.9365	0.0508	0.0894	0.0382	0.0478	0.916
+ PanBase ($\mathbf{s}_{\text{ms}}$)	–	–	–	–	–	–	–	–	–	–	–
TS-Pan [21]	38.53	0.9634	4.33	**3.48**	0.8674	0.9384	0.0535	0.0971	0.1402	0.1357	0.743
+ PanBase ($\mathbf{s}_{\text{pan}}$)	**38.54**	**0.9642**	**4.31**	**3.48**	**0.8697**	**0.9407**	0.0492	0.0869	0.0251	0.0341	0.942
+ PanBase ($\mathbf{s}_{\text{ms}}$)	38.49	0.9636	4.37	3.50	0.8652	0.9391	**0.0475**	0.0896	0.0095	0.1239	0.868
Mean change ($\mathbf{s}_{\text{pan}}$)	+0.89	+0.0091	−0.35	−0.50	+0.0289	+0.0128	−0.0124	−0.0126	−0.0108	−0.0221	+0.030
Mean change ($\mathbf{s}_{\text{ms}}$)	+1.30	+0.0137	−0.48	−0.72	+0.0424	+0.0178	−0.0181	−0.0158	−0.0192	−0.0073	+0.023

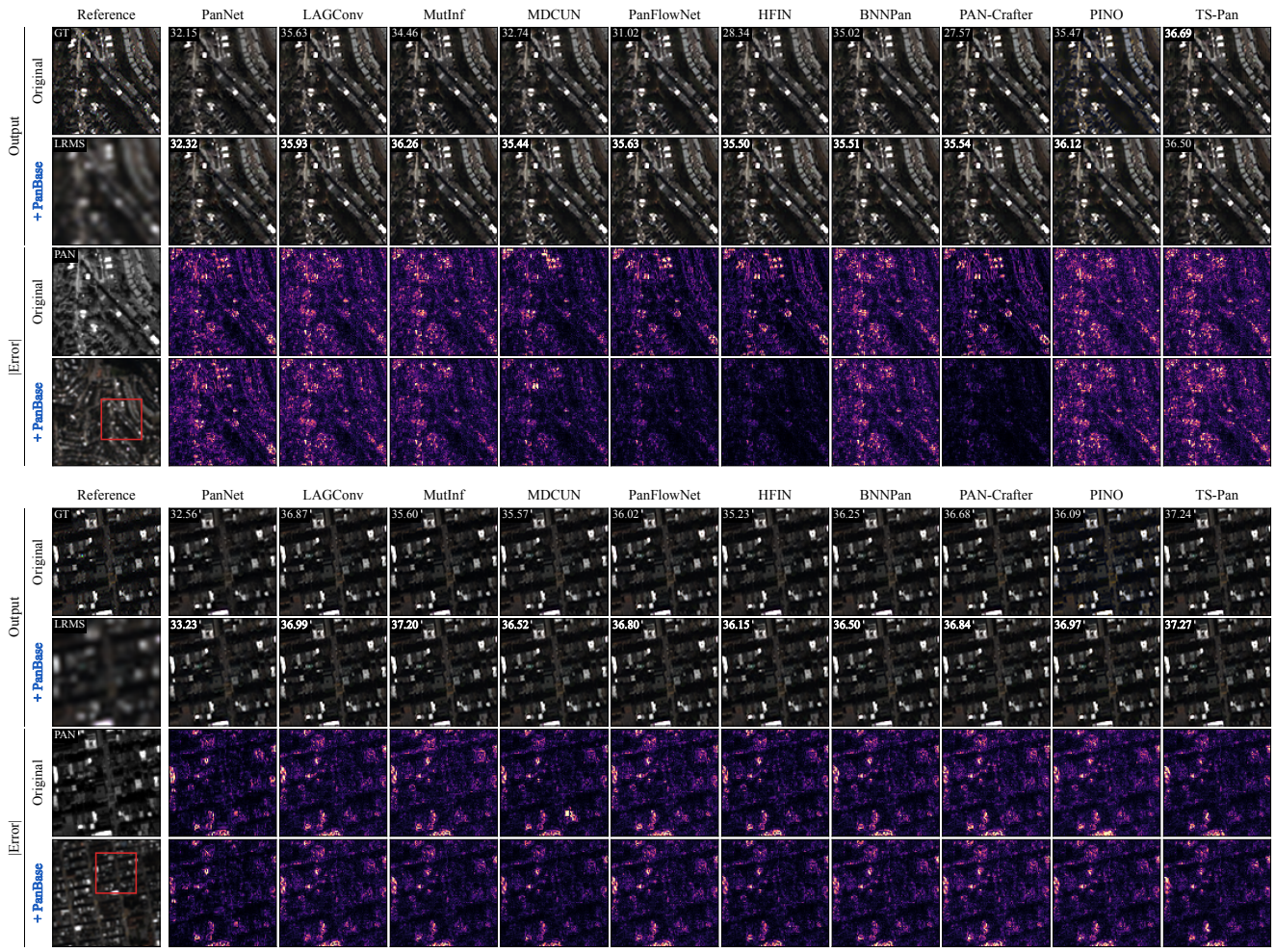

Figure 8. **Backbone reconstructions on QB at reduced resolution.** Each column pairs the original and PanBase outputs with their absolute error maps for two scenes. Scores are PSNR in dB. The HRMS column marks the crop used for comparison.

Table 12. **PanBase results on GF2.** Highlighted rows use PanBase as the residual base and an additional input. Green and red indicate gains and drops relative to each original backbone. **Bold** and underline mark the best and second-best scores per metric. Means exclude missing entries (–).

Method	Reduced resolution (HRMS reference)								Full resolution (no reference)		
	PSNR↑	SSIM↑	SAM↓	ERGAS↓	SCC↑	$Q2^n$↑	LPIPS↓	DISTS↓	D_λ↓	D_s↓	HQNR↑
PanNet [45]	36.87	0.9361	1.33	1.34	0.7032	0.9371	0.0846	0.1205	0.0041	0.0811	0.915
+ PanBase ($\mathbf{s}_{\text{pan}}$)	37.98	0.9460	1.15	1.19	0.7276	0.9499	0.0609	0.0957	0.0018	0.0711	0.927
+ PanBase ($\mathbf{s}_{\text{ms}}$)	40.06	0.9629	1.03	0.92	0.8412	0.9642	0.0537	0.0918	0.0012	0.0363	0.963
LAGConv [18]	42.47	0.9795	0.78	0.70	0.8921	0.9795	0.0252	0.0662	0.0041	0.0817	0.915
+ PanBase ($\mathbf{s}_{\text{pan}}$)	42.82	0.9806	0.76	0.67	0.8933	0.9811	0.0230	0.0622	0.0015	0.0496	0.949
+ PanBase ($\mathbf{s}_{\text{ms}}$)	43.02	0.9810	0.76	0.66	0.8964	0.9815	0.0228	0.0627	0.0008	0.0300	0.969
MutInf [47]	36.65	0.9545	1.35	1.39	0.8053	0.9413	0.0434	0.0962	0.0228	0.0355	0.943
+ PanBase ($\mathbf{s}_{\text{pan}}$)	43.59	0.9832	0.72	0.61	0.9100	0.9839	0.0186	0.0551	0.0015	0.0252	0.973
+ PanBase ($\mathbf{s}_{\text{ms}}$)	43.03	0.9814	0.74	0.65	0.9011	0.9817	0.0202	0.0572	0.0007	0.0153	0.984
MDCUN [43]	42.04	0.9768	0.81	0.74	0.8721	0.9783	0.0241	0.0625	0.0006	0.0467	0.953
+ PanBase ($\mathbf{s}_{\text{pan}}$)	42.57	0.9788	0.77	0.70	0.8813	0.9801	0.0213	0.0597	0.0004	0.0358	0.964
+ PanBase ($\mathbf{s}_{\text{ms}}$)	43.15	0.9818	0.74	0.65	0.8926	0.9826	0.0201	0.0573	0.0002	0.0152	0.985
PanFlowNet [44]	38.56	0.9619	1.28	1.09	0.8226	0.9593	0.0451	0.1001	0.0262	0.0724	0.904
+ PanBase ($\mathbf{s}_{\text{pan}}$)	40.12	0.9669	0.99	0.89	0.8366	0.9656	0.0366	0.0848	0.0094	0.0634	0.928
+ PanBase ($\mathbf{s}_{\text{ms}}$)	40.19	0.9674	0.97	0.87	0.8581	0.9640	0.0394	0.0916	0.0070	0.0598	0.934
MSDDN [15]	38.72	0.9664	1.15	1.07	0.8667	0.9486	0.0330	0.0812	0.0142	0.0416	0.945
+ PanBase ($\mathbf{s}_{\text{pan}}$)	41.80	0.9769	0.79	0.75	0.8685	0.9773	0.0237	0.0635	0.0010	0.0429	0.956
+ PanBase ($\mathbf{s}_{\text{ms}}$)	41.86	0.9767	0.80	0.74	0.8686	0.9777	0.0252	0.0676	0.0005	0.0281	0.971
HFIN [30]	38.08	0.9716	1.07	1.13	0.8615	0.9282	0.0364	0.0882	0.0392	0.0538	0.909
+ PanBase ($\mathbf{s}_{\text{pan}}$)	40.95	0.9711	0.95	0.87	0.8451	0.9729	0.0260	0.0678	0.0019	0.0786	0.920
+ PanBase ($\mathbf{s}_{\text{ms}}$)	42.56	0.9793	0.80	0.69	0.8879	0.9796	0.0230	0.0675	0.0009	0.0328	0.966
UTeRM [24]	41.89	0.9759	0.84	0.75	0.8713	0.9771	0.0286	0.0748	0.0024	0.0574	0.940
+ PanBase ($\mathbf{s}_{\text{pan}}$)	42.56	0.9785	0.80	0.70	0.8827	0.9793	0.0247	0.0698	0.0009	0.0327	0.966
+ PanBase ($\mathbf{s}_{\text{ms}}$)	42.69	0.9794	0.79	0.70	0.8832	0.9808	0.0226	0.0634	0.0002	0.0259	0.974
BNNPan [17]	42.12	0.9774	0.81	0.72	0.8813	0.9778	0.0271	0.0715	0.0036	0.0694	0.927
+ PanBase ($\mathbf{s}_{\text{pan}}$)	42.69	0.9791	0.77	0.68	0.8870	0.9799	0.0265	0.0697	0.0020	0.0557	0.942
+ PanBase ($\mathbf{s}_{\text{ms}}$)	–	–	–	–	–	–	–	–	–	–	–
U-Know-DiffPAN [20]	43.24	0.9824	0.68	0.64	0.8924	0.9833	0.0233	0.0594	0.0012	0.0325	0.966
+ PanBase ($\mathbf{s}_{\text{pan}}$)	44.05	0.9845	0.64	0.59	0.9054	0.9853	0.0213	0.0552	0.0005	0.0235	0.976
+ PanBase ($\mathbf{s}_{\text{ms}}$)	–	–	–	–	–	–	–	–	–	–	–
PAN-Crafter [10]	44.33	0.9857	0.65	0.57	0.9192	0.9860	0.0205	0.0565	0.0041	0.0254	0.971
+ PanBase ($\mathbf{s}_{\text{pan}}$)	44.80	0.9868	0.62	0.53	0.9224	0.9872	**0.0167**	**0.0524**	0.0013	0.0198	0.979
+ PanBase ($\mathbf{s}_{\text{ms}}$)	**45.00**	**0.9872**	**0.61**	**0.52**	**0.9235**	**0.9875**	0.0174	0.0526	0.0014	0.0179	0.981
PINO [22]	42.48	0.9808	0.91	0.72	0.9023	0.9811	0.0300	0.0717	0.0024	0.0185	0.979
+ PanBase ($\mathbf{s}_{\text{pan}}$)	44.20	0.9852	0.69	0.57	0.9136	0.9855	0.0204	0.0569	0.0011	**0.0125**	**0.986**
+ PanBase ($\mathbf{s}_{\text{ms}}$)	44.66	0.9860	0.65	0.54	0.9172	0.9864	0.0205	0.0597	0.0004	0.0161	0.984
PanDiT [12]	42.00	0.9767	0.80	0.74	0.8690	0.9786	0.0245	0.0698	0.0009	0.0449	0.954
+ PanBase ($\mathbf{s}_{\text{pan}}$)	42.16	0.9769	0.80	0.73	0.8692	0.9791	0.0236	0.0691	**0.0001**	0.0184	0.982
+ PanBase ($\mathbf{s}_{\text{ms}}$)	42.34	0.9777	0.80	0.71	0.8727	0.9794	0.0240	0.0674	**0.0001**	0.0220	0.978
MMMamba [38]	39.68	0.9484	1.23	1.11	0.8040	0.9457	0.0582	0.0882	0.0201	0.0535	0.928
+ PanBase ($\mathbf{s}_{\text{pan}}$)	42.72	0.9791	0.84	0.72	0.8864	0.9806	0.0193	0.0574	0.0063	0.0325	0.961
+ PanBase ($\mathbf{s}_{\text{ms}}$)	–	–	–	–	–	–	–	–	–	–	–
TS-Pan [21]	41.94	0.9761	0.81	0.74	0.8567	0.9742	0.0273	0.1073	0.0133	0.0430	0.944
+ PanBase ($\mathbf{s}_{\text{pan}}$)	44.34	0.9854	0.64	0.56	0.9182	0.9857	0.0250	0.0590	0.0095	0.0600	0.931
+ PanBase ($\mathbf{s}_{\text{ms}}$)	44.56	0.9859	0.63	0.55	0.9170	0.9858	0.0204	0.0548	0.0004	0.0210	0.979
Mean change ($\mathbf{s}_{\text{pan}}$)	+1.75	+0.0072	−0.17	−0.18	+0.0218	+0.0132	−0.0096	−0.0157	−0.0080	−0.0090	+0.017
Mean change ($\mathbf{s}_{\text{ms}}$)	+2.26	+0.0087	−0.20	−0.23	+0.0348	+0.0152	−0.0095	−0.0168	−0.0101	−0.0235	+0.033

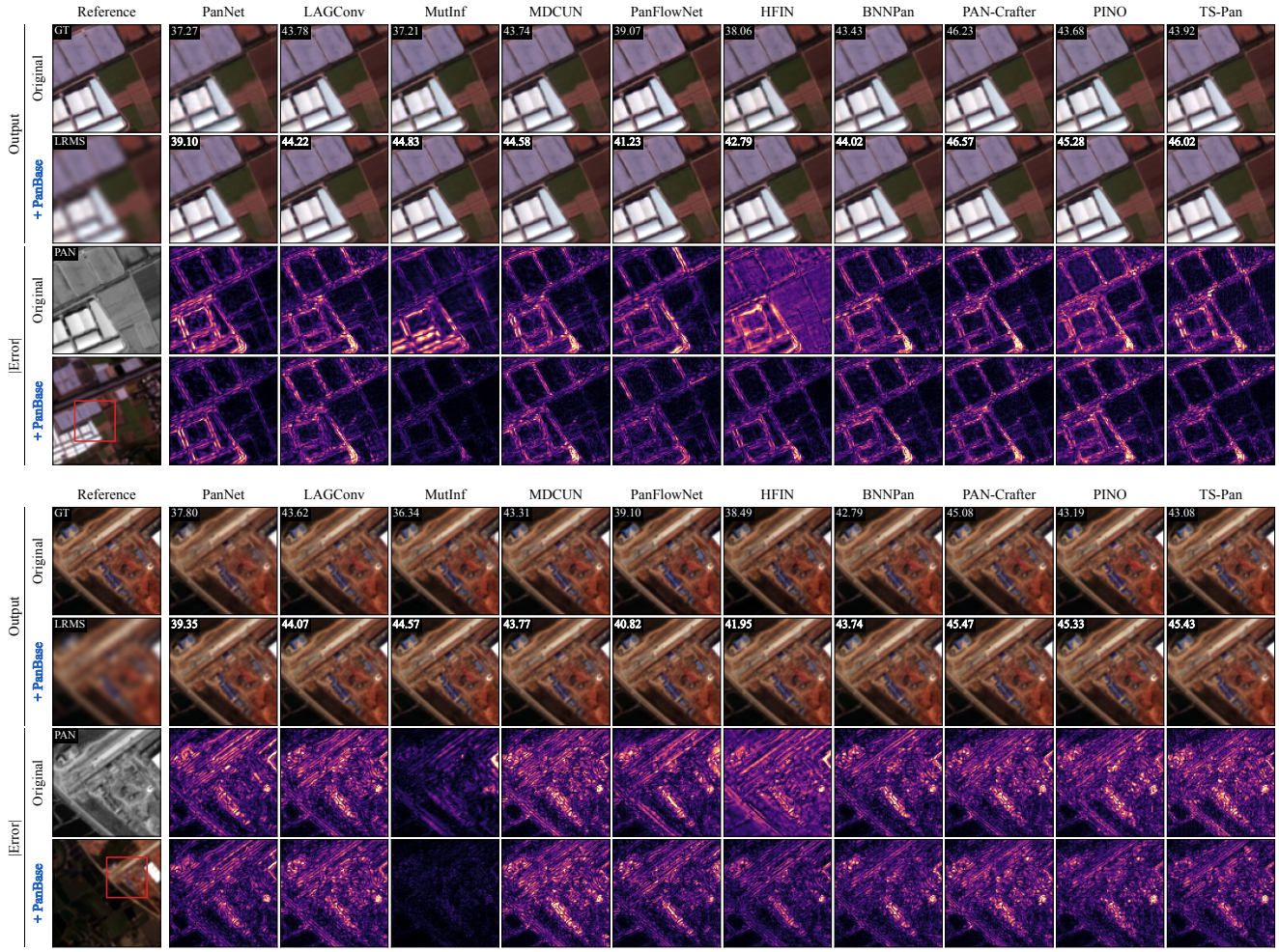

Figure 9. **Backbone reconstructions on GF2 at reduced resolution.** Each column pairs the original and PanBase outputs with their absolute error maps for two scenes. Scores are PSNR in dB. In the upper scene, MutInf improves from 37.21 to 44.83 dB with smaller errors around the bright roofs.

Table 13. **PanBase results on unseen WV2.** Highlighted rows use PanBase as the residual base and an additional input. Green and red indicate gains and drops relative to each original backbone. **Bold** and underline mark the best and second-best scores per metric. Means exclude missing entries (–). WV3-trained models transfer without fine-tuning. The standalone bases use WV3 coefficients with the WV2 sensor operator.

Method	Reduced resolution (HRMS reference)								Full resolution (no reference)		
	PSNR↑	SSIM↑	SAM↓	ERGAS↓	SCC↑	$Q2^n$↑	LPIPS↓	DISTS↓	D_λ↓	D_s↓	HQNR↑
PanNet [45]	28.51	0.8095	5.81	4.64	0.6885	0.8230	0.2294	0.1866	0.0169	0.0158	0.968
+ PanBase ($\mathbf{s}_{\text{pan}}$)	29.10	0.8277	5.52	4.34	0.6765	0.8360	0.1885	0.1686	0.0054	0.0371	0.958
+ PanBase ($\mathbf{s}_{\text{ms}}$)	29.40	0.8384	5.48	4.17	0.7036	0.8434	0.1525	0.1550	0.0061	0.0312	0.963
LAGConv [18]	28.71	0.8195	5.61	4.55	0.6603	0.8317	0.2178	0.1769	0.0244	0.0275	0.949
+ PanBase ($\mathbf{s}_{\text{pan}}$)	29.17	0.8341	5.48	4.34	0.6327	0.8410	0.2059	0.1745	0.0120	0.0857	0.903
+ PanBase ($\mathbf{s}_{\text{ms}}$)	29.07	0.8324	5.51	4.36	0.6522	0.8434	0.2038	0.1715	0.0129	0.0512	0.936
MutInf [47]	28.87	0.8399	5.47	4.49	0.6737	0.8306	0.1602	0.1388	0.0634	0.0371	0.902
+ PanBase ($\mathbf{s}_{\text{pan}}$)	29.01	0.8234	5.28	4.42	0.6073	0.8356	0.2412	0.1857	0.0083	0.0521	0.940
+ PanBase ($\mathbf{s}_{\text{ms}}$)	29.23	0.8351	5.18	4.31	0.6364	0.8417	0.2075	0.1618	0.0095	0.0558	0.935
MDCUN [43]	28.96	0.8263	5.40	4.45	0.6937	0.8368	0.2318	0.1892	0.0156	0.0473	0.938
+ PanBase ($\mathbf{s}_{\text{pan}}$)	30.05	0.8638	5.28	3.84	0.7206	0.8687	0.1579	0.1377	0.0029	0.0359	0.961
+ PanBase ($\mathbf{s}_{\text{ms}}$)	28.79	0.8139	5.58	4.53	0.6160	0.8247	0.2632	0.1891	0.0096	0.0501	0.941
PanFlowNet [44]	27.78	0.8060	6.17	5.16	0.6582	0.7883	0.2700	0.1887	0.0745	0.0609	0.869
+ PanBase ($\mathbf{s}_{\text{pan}}$)	28.19	0.8026	5.96	4.84	0.6355	0.8085	0.2491	0.1940	0.0256	0.0393	0.936
+ PanBase ($\mathbf{s}_{\text{ms}}$)	28.22	0.7970	5.80	4.86	0.6202	0.8067	0.2595	0.1949	0.0370	0.0545	0.912
MSDDN [15]	28.65	0.8304	5.73	4.59	0.6770	0.8235	0.1853	0.1715	0.0426	0.0279	0.931
+ PanBase ($\mathbf{s}_{\text{pan}}$)	28.82	0.8237	5.47	4.51	0.6085	0.8338	0.2094	0.1734	0.0101	0.0710	0.920
+ PanBase ($\mathbf{s}_{\text{ms}}$)	28.68	0.8270	5.57	4.55	0.6229	0.8367	0.1807	0.1721	0.0138	0.0881	0.899
HFIN [30]	28.39	0.8135	5.87	4.75	0.6967	0.8123	0.2461	0.1907	0.0504	0.0488	0.903
+ PanBase ($\mathbf{s}_{\text{pan}}$)	29.47	0.8413	5.23	4.18	0.6799	0.8490	0.2020	0.1659	0.0076	0.0411	0.952
+ PanBase ($\mathbf{s}_{\text{ms}}$)	28.68	0.8078	5.36	4.63	0.6101	0.8239	0.2691	0.2036	0.0137	0.0650	0.922
UTeRM [24]	28.85	0.8347	5.73	4.48	0.7300	0.8350	0.2173	0.1730	0.0319	0.0144	0.954
+ PanBase ($\mathbf{s}_{\text{pan}}$)	29.13	0.8366	5.62	4.33	0.6739	0.8447	0.2311	0.1844	0.0300	0.0492	0.922
+ PanBase ($\mathbf{s}_{\text{ms}}$)	29.45	0.8514	5.50	4.18	0.7198	0.8494	0.1994	0.1694	0.0225	0.0326	0.946
BNNPan [17]	27.35	0.7740	7.10	5.40	0.4945	0.7837	0.2420	0.2196	0.1584	0.1241	0.755
+ PanBase ($\mathbf{s}_{\text{pan}}$)	29.69	0.8604	5.46	4.00	0.7059	0.8614	0.1358	0.1308	0.0244	0.0274	0.949
+ PanBase ($\mathbf{s}_{\text{ms}}$)	–	–	–	–	–	–	–	–	–	–	–
U-Know-DiffPAN [20]	29.20	0.8350	5.22	4.28	0.6857	0.8466	0.2267	0.1817	0.0154	0.0192	0.966
+ PanBase ($\mathbf{s}_{\text{pan}}$)	29.80	0.8539	5.07	4.03	0.6895	0.8595	0.2165	0.1763	0.0092	0.0440	0.947
+ PanBase ($\mathbf{s}_{\text{ms}}$)	30.04	0.8614	4.98	3.89	0.7314	0.8650	0.2003	0.1664	0.0066	0.0341	0.960
PAN-Crafter [10]	29.04	0.8267	5.17	4.42	0.6698	0.8378	0.2783	0.1977	0.0236	0.0286	0.949
+ PanBase ($\mathbf{s}_{\text{pan}}$)	29.90	0.8551	5.08	3.94	0.7389	0.8621	0.2092	0.1735	0.0133	0.0505	0.937
+ PanBase ($\mathbf{s}_{\text{ms}}$)	28.88	0.8179	5.26	4.49	0.6259	0.8326	0.2939	0.2035	0.0105	0.0568	0.933
PINO [22]	29.76	0.8584	5.11	4.00	0.7295	0.8595	0.2154	0.1706	0.0270	0.0273	0.946
+ PanBase ($\mathbf{s}_{\text{pan}}$)	30.14	0.8636	4.87	3.84	0.7515	0.8675	0.2086	0.1613	0.0070	0.0400	0.953
+ PanBase ($\mathbf{s}_{\text{ms}}$)	**30.43**	0.8718	**4.83**	3.73	**0.7517**	0.8732	0.1993	0.1624	0.0069	0.0522	0.941
PanDiT [12]	29.15	0.8310	5.33	4.32	0.6904	0.8383	0.2087	0.1748	0.0098	**0.0123**	**0.978**
+ PanBase ($\mathbf{s}_{\text{pan}}$)	30.17	0.8680	5.15	3.80	0.7155	0.8713	0.1510	0.1322	0.0009	0.0368	0.962
+ PanBase ($\mathbf{s}_{\text{ms}}$)	30.37	0.8724	5.11	3.69	0.7325	0.8743	0.1374	0.1253	0.0010	0.0375	0.962
MMMamba [38]	28.89	0.8224	5.46	4.45	0.6199	0.8372	0.2007	0.1785	0.0129	0.0554	0.932
+ PanBase ($\mathbf{s}_{\text{pan}}$)	28.88	0.8197	5.41	4.47	0.5611	0.8311	0.2298	0.1752	0.0093	0.0920	0.899
+ PanBase ($\mathbf{s}_{\text{ms}}$)	27.96	0.7975	5.84	5.00	0.4591	0.8151	0.1976	0.1960	0.0115	0.1128	0.877
TS-Pan [21]	29.59	0.8473	5.12	4.10	0.6941	0.8559	0.2025	0.1685	0.0104	0.0705	0.920
+ PanBase ($\mathbf{s}_{\text{pan}}$)	30.10	0.8641	5.13	3.84	0.6932	0.8688	0.1760	0.1550	0.0030	0.0672	0.930
+ PanBase ($\mathbf{s}_{\text{ms}}$)	30.15	0.8625	5.01	3.82	0.6805	0.8672	0.1644	0.2004	0.0055	0.0596	0.935
Mean change ($\mathbf{s}_{\text{pan}}$)	+0.66	+0.0175	−0.29	−0.36	+0.0019	+0.0199	−0.0213	−0.0146	−0.0272	+0.0101	+0.014
Mean change ($\mathbf{s}_{\text{ms}}$)	+0.36	+0.0061	−0.16	−0.18	−0.0289	+0.0101	−0.0116	−0.0011	−0.0180	+0.0206	−0.003
Standalone PanBase ($\mathbf{s}_{\text{pan}}$)	30.12	0.8691	5.49	3.77	0.7319	0.8741	0.1323	0.1263	**0.0005**	0.0404	0.959
Standalone PanBase ($\mathbf{s}_{\text{ms}}$)	30.34	**0.8747**	5.48	**3.65**	0.7503	**0.8784**	**0.1203**	**0.1212**	**0.0005**	0.0517	0.948

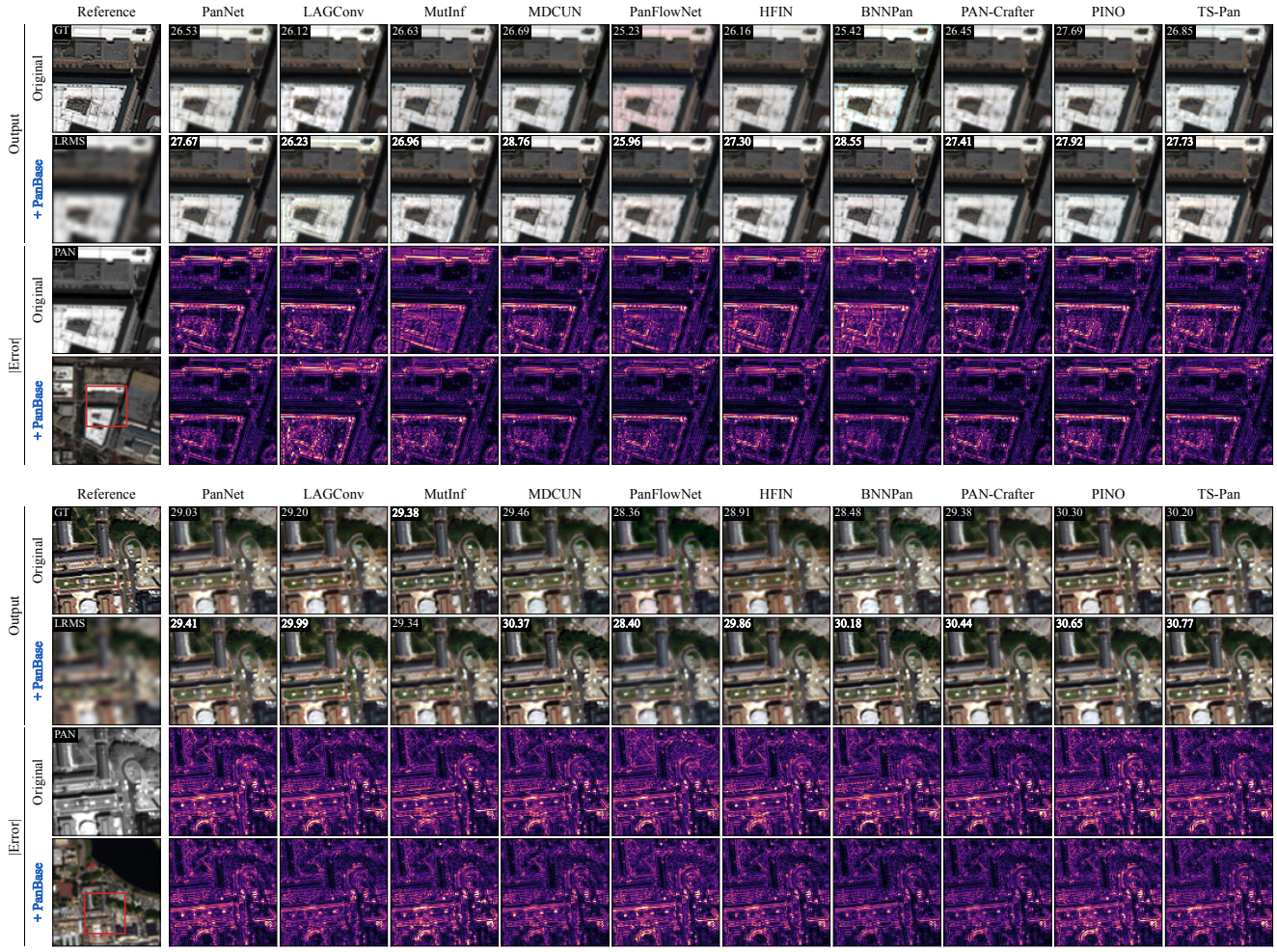

Figure 10. **Backbone reconstructions on unseen WV2 at reduced resolution.** Each column pairs the original and PanBase outputs with their absolute error maps for two scenes. Scores are PSNR in dB. The models and calibration coefficients come from WV3. The upper scene shows BNNPan improving from 25.42 to 28.55 dB.

Table 14. **Classical pan-sharpening versus standalone PanBase on WV3.** All methods operate without neural networks. PanBase coefficients are calibrated once using HRMS training pairs. CS/MRA/VO denote component substitution, multiresolution analysis, and variational optimization. Bold and underline indicate the best and second-best values per metric.

Type	Method	Reduced resolution (RR)								Full resolution (FR)		
		PSNR↑	SSIM↑	SAM↓	ERGAS↓	SCC↑	$Q2^n$ ↑	LPIPS↓	DISTS↓	D_λ ↓	D_s ↓	HQNR↑
–	LRMS↑ (interpolated) [1]	27.12	0.7130	5.80	7.16	0.1678	0.6272	0.5881	0.3175	0.0155	0.0772	0.909
CS	BT-H [23]	31.22	0.8947	4.92	4.58	0.6614	0.8327	0.1650	0.1575	0.0308	0.0811	0.891
	BDSD-PC [31]	30.96	0.8890	5.43	4.70	0.6400	0.8296	0.1591	0.1502	0.0360	0.0727	0.895
	GSA [3]	30.78	0.8821	5.37	4.77	0.6387	0.8273	0.1704	0.1578	0.0271	0.0794	0.896
	PRACS [7]	29.95	0.8559	5.58	5.25	0.6285	0.7899	0.2056	0.1704	0.0206	0.0453	0.935
MRA	AWLP [25]	31.21	0.8902	5.23	4.66	0.6492	0.8352	0.1753	0.1646	0.0105	0.0524	0.938
	MTF-GLP-FS [34]	30.91	0.8837	5.32	4.70	0.6386	0.8335	0.1735	0.1602	0.0089	0.0627	0.929
	MTF-GLP-HPM [2, 5]	31.38	0.8960	5.13	4.50	0.6547	0.8384	0.1593	0.1531	0.0104	0.0702	0.920
	MF [29]	30.56	0.8786	5.26	4.86	0.6481	0.8220	0.1935	0.1845	0.0171	0.0653	0.919
VO	TV [26]	30.48	0.8801	5.69	4.86	0.6330	0.7955	0.1510	0.1546	0.0095	0.0386	0.952
	FE-HPM [33]	30.87	0.8821	5.23	4.67	0.6549	0.8325	0.1791	0.1731	0.0109	0.0667	0.923
	PWMBF [27]	30.07	0.8699	5.89	5.14	0.6359	0.8056	0.1968	0.1779	0.0330	0.0695	0.900
	LRTCFPan [41]	31.74	0.9024	4.74	4.32	0.6452	0.8467	0.1720	0.1521	0.0054	0.0525	0.942
Ours	**PanBase** ($\mathbf{s}_{\text{pan}}$)	32.28	0.9147	4.65	4.07	0.6608	0.8561	0.0806	0.1136	**0.0005**	**0.0365**	**0.963**
	PanBase ($\mathbf{s}_{\text{ms}}$)	**33.88**	**0.9402**	**4.62**	**3.30**	**0.7917**	**0.8830**	**0.0663**	**0.1014**	0.0006	0.0841	0.915

Table 15. **Classical pan-sharpening versus standalone PanBase on QB.** All methods operate without neural networks. PanBase coefficients are calibrated once using HRMS training pairs. CS/MRA/VO denote component substitution, multiresolution analysis, and variational optimization. Bold and underline indicate the best and second-best values per metric.

Type	Method	Reduced resolution (RR)								Full resolution (FR)		
		PSNR↑	SSIM↑	SAM↓	ERGAS↓	SCC↑	$Q2^n$ ↑	LPIPS↓	DISTS↓	D_λ ↓	D_s ↓	HQNR↑
–	LRMS↑ (interpolated) [1]	28.06	0.6915	8.44	11.82	0.1681	0.5839	0.6627	0.3384	0.0273	0.1456	0.831
CS	BT-H [23]	32.09	0.8695	7.23	7.34	0.6459	0.8386	0.1587	0.1805	0.1143	0.2273	0.685
	BDSD-PC [31]	31.89	0.8603	8.09	7.51	0.6298	0.8305	0.1766	0.1843	0.1020	0.1659	0.750
	GSA [3]	31.75	0.8554	8.10	7.62	0.6293	0.8269	0.1942	0.1895	0.1120	0.2270	0.687
	PRACS [7]	31.00	0.8347	8.19	8.32	0.6290	0.7889	0.2244	0.1950	0.0651	0.1649	0.781
MRA	AWLP [25]	31.89	0.8575	8.19	7.58	0.6386	0.8292	0.1814	0.1735	0.0238	0.1427	0.837
	MTF-GLP-FS [34]	32.02	0.8613	7.79	7.37	0.6304	0.8349	0.1879	0.1843	0.0220	0.1518	0.830
	MTF-GLP-HPM [2, 5]	32.24	0.8696	7.79	7.22	0.6441	0.8454	0.1656	0.1765	0.0262	0.1519	0.826
	MF [29]	31.47	0.8458	7.93	7.87	0.6364	0.8185	0.2059	0.2089	0.0348	0.1397	0.831
VO	TV [26]	31.59	0.8561	7.51	7.69	0.6255	0.8214	0.2202	0.1831	0.0254	0.1020	0.875
	FE-HPM [33]	31.71	0.8495	7.95	7.65	0.6437	0.8266	0.1943	0.2010	0.0266	0.1517	0.826
	PWMBF [27]	31.00	0.8328	8.87	8.28	0.6266	0.7892	0.2303	0.2155	0.0686	0.1604	0.782
	LRTCFPan [41]	32.57	0.8749	7.19	6.93	0.6349	0.8553	0.1780	0.1725	0.0042	0.0701	0.926
Ours	**PanBase** ($\mathbf{s}_{\text{pan}}$)	33.10	0.8909	6.56	6.53	0.6470	0.8707	0.1157	0.1495	0.0033	**0.0525**	**0.944**
	PanBase ($\mathbf{s}_{\text{ms}}$)	**36.05**	**0.9311**	**6.11**	**4.63**	**0.8098**	**0.9130**	**0.0887**	**0.1399**	**0.0028**	0.2953	0.703

Table 16. **Classical pan-sharpening versus standalone PanBase on GF2.** All methods operate without neural networks. PanBase coefficients are calibrated once using HRMS training pairs. CS/MRA/VO denote component substitution, multiresolution analysis, and variational optimization. Bold and underline indicate the best and second-best values per metric.

Type	Method	Reduced resolution (RR)								Full resolution (FR)		
		PSNR↑	SSIM↑	SAM↓	ERGAS↓	SCC↑	$Q2^n$ ↑	LPIPS↓	DISTS↓	D_λ ↓	D_s ↓	HQNR↑
–	LRMS↑ (interpolated) [1]	32.08	0.8239	1.82	2.37	0.3183	0.8125	0.4409	0.2764	0.0072	0.0885	0.905
CS	BT-H [23]	35.80	0.9239	1.56	1.57	0.7232	0.9176	0.0868	0.1163	0.0232	0.1481	0.832
	BDSD-PC [31]	35.10	0.9059	1.68	1.67	0.6918	0.8924	0.1123	0.1243	0.0200	0.1540	0.829
	GSA [3]	34.76	0.8961	1.72	1.72	0.6902	0.8864	0.1267	0.1320	0.0288	0.1846	0.792
	PRACS [7]	35.32	0.9132	1.69	1.63	0.6987	0.9046	0.1257	0.1310	0.0151	0.1221	0.865
MRA	AWLP [25]	34.45	0.8846	1.94	1.64	0.6903	0.8731	0.1358	0.1491	0.0092	0.1352	0.857
	MTF-GLP-FS [34]	35.43	0.9021	1.66	1.59	0.6915	0.8968	0.1253	0.1317	0.0056	0.1419	0.853
	MTF-GLP-HPM [2, 5]	34.71	0.8893	1.65	1.72	0.6979	0.8846	0.1221	0.1351	0.0071	0.1527	0.841
	MF [29]	34.71	0.9018	1.64	1.70	0.7053	0.8900	0.1140	0.1322	0.0154	0.1575	0.830
VO	TV [26]	35.04	0.9149	1.91	1.74	0.6979	0.9072	0.0937	0.1154	0.0096	0.1107	0.881
	FE-HPM [33]	35.39	0.9063	1.61	1.58	0.7059	0.8988	0.1107	0.1289	0.0134	0.1666	0.822
	PWMBF [27]	33.87	0.8966	1.96	1.83	0.6814	0.8631	0.1627	0.1622	0.0196	0.1015	0.881
	LRTCFPan [41]	37.53	0.9335	1.30	1.27	0.6901	0.9351	0.0905	0.1213	0.0027	0.0906	0.907
Ours	**PanBase** ($\mathbf{s}_{\text{pan}}$)	38.31	0.9446	1.21	1.17	0.7352	0.9503	0.0898	0.1168	**0.0000**	0.0350	0.965
	PanBase ($\mathbf{s}_{\text{ms}}$)	**39.86**	**0.9584**	**1.09**	**0.94**	**0.8232**	**0.9605**	**0.0660**	**0.1000**	**0.0000**	**0.0294**	**0.971**

Table 17. **Classical pan-sharpening versus standalone PanBase on unseen WV2.** All methods operate without neural networks. PanBase coefficients are calibrated once using HRMS training pairs. CS/MRA/VO denote component substitution, multiresolution analysis, and variational optimization. PanBase reuses WV3 coefficients with the WV2 sensor operator. Bold and underline indicate the best and second-best values per metric.

Type	Method	Reduced resolution (RR)								Full resolution (FR)		
		PSNR↑	SSIM↑	SAM↓	ERGAS↓	SCC↑	$Q2^n$ ↑	LPIPS↓	DISTS↓	D_λ ↓	D_s ↓	HQNR↑
–	LRMS↑ (interpolated) [1]	25.38	0.6160	6.53	6.77	0.1582	0.6465	0.6558	0.3293	0.0236	0.0559	0.922
CS	BT-H [23]	28.90	0.8319	5.90	4.41	0.7322	0.8364	0.2284	0.1803	0.0379	0.0866	0.879
	BDSD-PC [31]	29.04	0.8399	6.14	4.26	0.7261	0.8520	0.2016	0.1577	0.0760	0.0396	0.888
	GSA [3]	28.51	0.8159	6.25	4.50	0.7180	0.8358	0.2214	0.1697	0.0376	0.0887	0.877
	PRACS [7]	27.24	0.7554	6.32	5.31	0.6860	0.7756	0.2804	0.2039	0.0264	0.0332	0.941
MRA	AWLP [25]	28.87	0.8305	6.07	4.44	0.7272	0.8390	0.2361	0.1872	0.0160	0.0686	0.917
	MTF-GLP-FS [34]	28.59	0.8185	6.19	4.46	0.7183	0.8379	0.2249	0.1722	0.0144	0.0762	0.911
	MTF-GLP-HPM [2, 5]	29.10	0.8409	6.06	4.23	0.7287	0.8507	0.2122	0.1688	0.0181	0.0795	0.904
	MF [29]	28.41	0.8189	6.13	4.61	0.7158	0.8333	0.2508	0.1948	0.0228	0.0812	0.898
VO	TV [26]	28.25	0.8150	6.03	4.71	0.7082	0.8260	0.2070	0.1561	0.0237	**0.0225**	0.954
	FE-HPM [33]	28.71	0.8234	6.10	4.43	0.7246	0.8382	0.2361	0.1841	0.0189	0.0811	0.902
	PWMBF [27]	27.60	0.7980	7.08	4.98	0.7200	0.8052	0.2355	0.1814	0.0455	0.1220	0.839
	LRTCFPan [41]	29.38	0.8436	5.62	4.12	0.6983	0.8565	0.2293	0.1603	0.0075	0.0575	0.935
Ours	**PanBase** ($\mathbf{s}_{\text{pan}}$)	30.12	0.8691	5.49	3.77	0.7319	0.8741	0.1323	0.1263	**0.0005**	0.0404	**0.959**
	PanBase ($\mathbf{s}_{\text{ms}}$)	**30.34**	**0.8747**	**5.48**	**3.65**	**0.7503**	**0.8784**	**0.1203**	**0.1212**	**0.0005**	0.0517	0.948

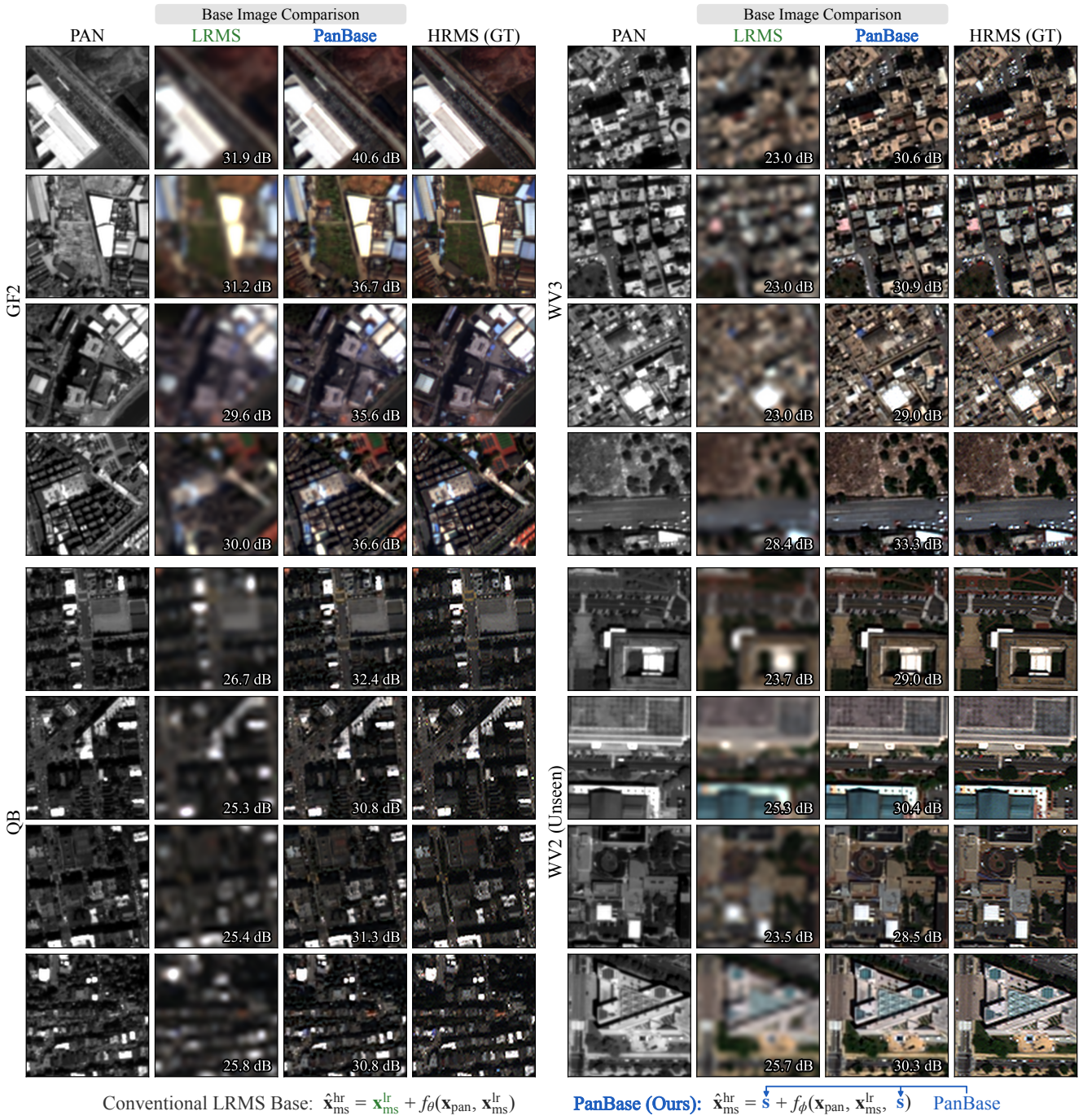

Figure 11. **Residual bases before neural refinement.** Each crop shows the observed PAN, interpolated LRMS, PanBase, and reference HRMS. The scores compare the two bases against HRMS in dB. The examples cover GF2, QB, WV3, and unseen WV2.



Figure 12. **Standalone reconstruction on four reduced-resolution scenes.** Each sensor panel compares the two PanBase geometries with 12 classical methods. The HRMS and interpolated LRMS provide references. Insets show reconstruction errors, and scores give per-scene PSNR in dB.

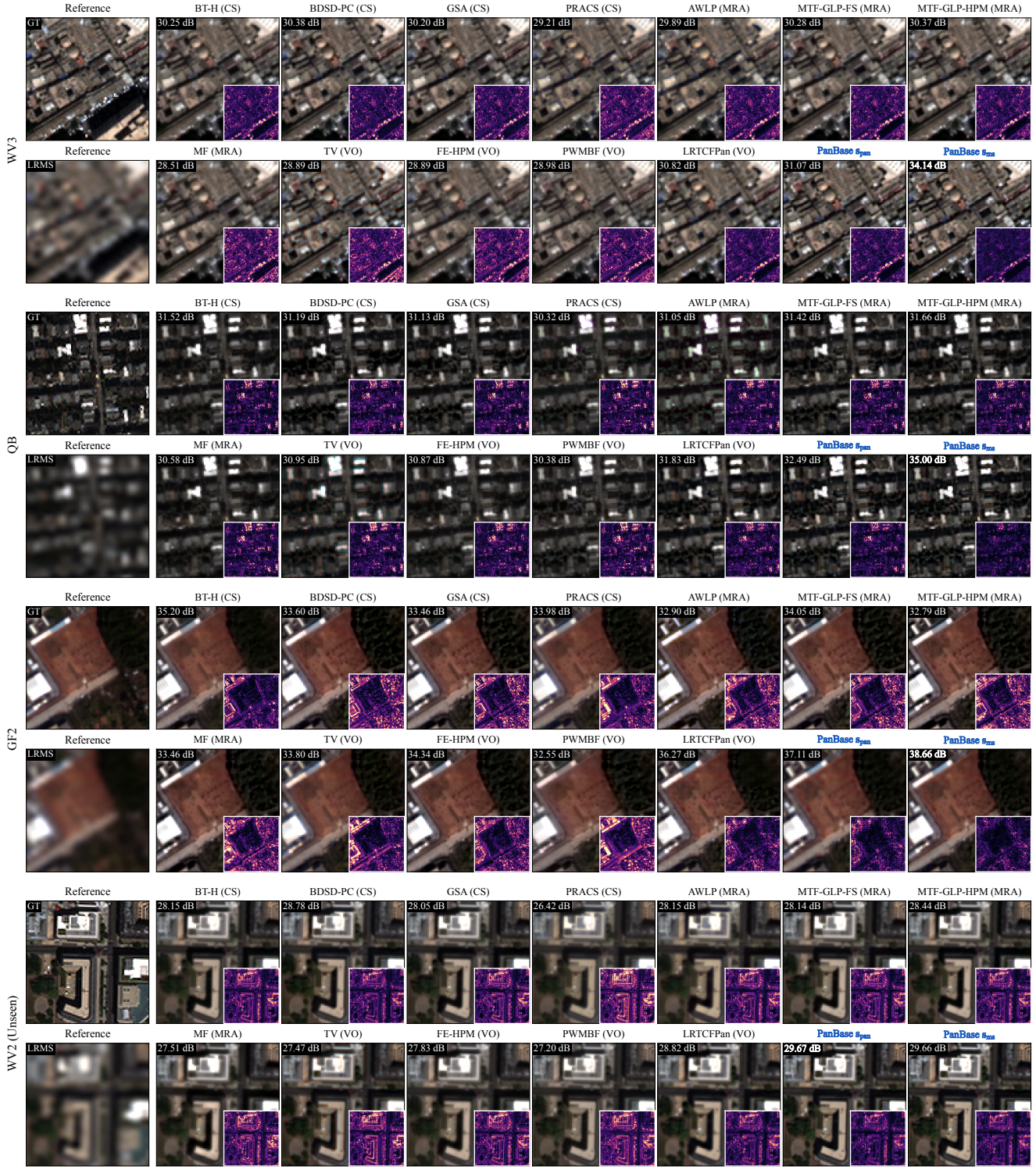

Figure 13. **Additional scenes for standalone fusion.** In the GF2 panel, PAN- and MS-aligned PanBase reach 37.11 and 38.66 dB. In the unseen WV2 panel, they reach 29.67 and 29.66 dB using coefficients calibrated on WV3. Insets show the corresponding errors against HRMS.

Table 18. **Ablation of PanBase construction in MS geometry.** Here $\mathbf{d}$ denotes the PAN detail after alignment to the MS geometry by the input-estimated translation τ. Network-free bases add components cumulatively, with coefficients recalibrated per row and projection applied after detail injection. Coeff. counts calibrated scalars for C MS bands. The highlighted row is the complete $\mathbf{s}_{\mathrm{ms}}$. PSNR/SAM use RR and HQNR uses FR. WV2 reuses WV3 coefficients with its own sensor operator.

Configuration	**Coeff.**	**WV3**			**QB**			**GF2**			**WV2** (Unseen)		
		PSNR↑	SAM↓	HQNR↑	PSNR↑	SAM↓	HQNR↑	PSNR↑	SAM↓	HQNR↑	PSNR↑	SAM↓	HQNR↑
Interpolated LRMS ($\mathbf{x}^{\mathrm{lr}}_{\mathrm{ms},b}$)	0	27.12	5.80	0.909	28.06	8.44	0.831	32.08	1.82	0.905	25.38	6.53	0.922
MS geometry ($\mathbf{d}$ shifted by τ)													
+ Projection ($\Pi_b(\cdot)$)	0	27.99	5.44	0.927	29.09	7.86	0.822	34.21	1.47	0.951	26.20	6.05	0.901
+ Additive detail ($\mathbf{r}^1_b = \mathbf{d}$)	C	32.95	5.21	0.923	35.30	6.97	0.710	39.01	1.20	0.970	29.65	5.62	0.949
+ Multiplicative detail ($\mathbf{r}^2_b = \mathbf{m}_b \odot \mathbf{d}$)	$2C$	33.55	4.86	0.918	35.83	6.20	0.705	39.27	1.12	0.969	30.14	5.56	0.949
+ Local regression ($\mathbf{r}^3_b = \boldsymbol{\beta}_b \odot \mathbf{d}$)	$3C$	33.81	4.63	0.916	35.94	6.11	0.704	39.59	1.10	0.967	30.30	5.52	0.947
+ Frequency bands ($\mathbf{r}^{i,j}_b = F_j(\mathbf{r}^i_b)$)	$9C$	33.88	4.62	0.915	36.05	6.11	0.703	39.86	1.09	0.971	30.34	5.48	0.948

Table 19. **Ablation of PanBase integration in MS geometry.** With $\mathbf{s}_{\mathrm{ms}}$, we enable skip replacement, input concatenation, or both under matched training. Original retains the interpolated-LRMS skip and original inputs. Mean change averages the six backbone–dataset pairs.

Backbone	**Dataset**	**RR PSNR↑ (dB)**				**FR HQNR↑**			
		Original	Skip only	Input only	**Both (PanBase)**	Original	Skip only	Input only	**Both (PanBase)**
PanNet [45]	WV3	34.29	33.95 −0.34	35.12 +0.83	35.22 +0.93	0.934	0.919 −0.015	0.963 +0.029	0.967 +0.034
	QB	33.83	35.79 +1.96	36.30 +2.47	36.41 +2.57	0.843	0.747 −0.096	0.918 +0.075	0.881 +0.038
	GF2	36.87	40.03 +3.15	39.79 +2.92	40.06 +3.18	0.915	0.970 +0.055	0.966 +0.051	0.963 +0.047
PINO [22]	WV3	36.42	37.35 +0.93	36.72 +0.30	37.55 +1.13	0.959	0.961 +0.003	0.964 +0.005	0.954 −0.005
	QB	37.37	38.18 +0.82	37.96 +0.59	38.03 +0.66	0.925	0.936 +0.010	0.934 +0.009	0.945 +0.020
	GF2	42.48	44.51 +2.04	42.88 +0.40	44.66 +2.19	0.979	0.984 +0.005	0.986 +0.007	0.984 +0.004
Mean change		–	+1.43	+1.25	+1.78	–	−0.006	+0.029	+0.023

Table 20. **Ablation of alternative residual bases.** Alternative bases share the same skip replacement, input concatenation, and training.

Base	**PanNet** [45], **WV3**		**PanNet** [45], **GF2**		**LAGConv** [18], **WV3**		**LAGConv** [18], **GF2**	
	PSNR↑	HQNR↑	PSNR↑	HQNR↑	PSNR↑	HQNR↑	PSNR↑	HQNR↑
LRMS↑ (Original)	34.29	0.934	36.87	0.915	37.22	0.944	42.47	0.915
Π(LRMS↑)	34.75 +0.46	0.956 +0.022	37.62 +0.74	0.915 -0.000	**37.35** +0.13	0.956 +0.012	42.90 +0.43	0.918 +0.004
GSA [3] (CS)	34.30 +0.01	0.915 −0.019	35.69 −1.19	0.839 −0.076	37.14 −0.07	0.955 +0.011	42.01 −0.46	0.908 −0.006
MTF-GLP-HPM [2, 5] (MRA)	34.81 +0.52	0.931 −0.003	36.92 +0.05	0.885 −0.031	37.17 −0.05	0.944 -0.000	42.36 −0.10	0.921 +0.007
LRTCFPan [41] (VO)	34.81 +0.52	0.942 +0.009	37.80 +0.93	0.904 −0.011	37.24 +0.02	0.956 +0.013	42.24 −0.23	0.948 +0.034
PanBase ($\mathbf{s}_{\mathrm{ms}}$)	**35.22** +0.93	**0.967** +0.034	**40.06** +3.18	**0.963** +0.047	37.29 +0.07	**0.962** +0.018	**43.02** +0.55	**0.969** +0.055

References

[1] 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. 6, 22, 23

[2] Bruno Aiazzi, Luciano 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, 4, 6, 7, 9, 22, 23, 27

[3] Bruno Aiazzi, Stefano Baronti, and Massimo Selva. Improving component substitution pansharpening through multivariate regression of MS+Pan data. *IEEE Transactions on Geoscience and Remote Sensing*, 45(10):3230–3239, 2007. 2, 6, 7, 22, 23, 27

[4] Luciano Alparone, Bruno Aiazzi, Stefano Baronti, Andrea Garzelli, Filippo Nencini, and Massimo Selva. Multispectral and panchromatic data fusion assessment without reference. *Photogrammetric Engineering & Remote Sensing*, 74 (2):193–200, 2008. 10

[5] Luciano Alparone, Andrea Garzelli, and Gemine Vivone. Intersensor statistical matching for pansharpening: Theoretical issues and practical solutions. *IEEE Transactions on Geoscience and Remote Sensing*, 55(8):4682–4695, 2017. 2, 6, 7, 22, 23, 27

[6] Sung-Jin Cho, Seo-Won Ji, Jun-Pyo Hong, Seung-Won Jung, and Sung-Jea Ko. Rethinking coarse-to-fine approach in single image deblurring. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4641–4650, 2021. 2, 3

[7] Jaewan Choi, Kiyun Yu, and Yongil Kim. A new adaptive component-substitution-based satellite image fusion by using partial replacement. *IEEE Transactions on Geoscience and Remote Sensing*, 49(1):295–309, 2011. 2, 6, 22, 23

[8] 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, 2021. 3

[9] 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. 5, 9, 10

[10] Jeonghyeok Do, Sungpyo Kim, Geunhyuk Youk, Jaehyup Lee, and Munchurl Kim. PAN-Crafter: Learning modality-consistent alignment for PAN-sharpening. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4242–4252, 2025. 2, 3, 5, 7, 11, 12, 14, 16, 18, 20

[11] Yule Duan, Xiao Wu, Haoyu Deng, and Liang-Jian Deng. Content-adaptive non-local convolution for remote sensing pansharpening. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27738–27747, 2024. 2

[12] Jiabin Fang, Ke Cao, Xuanhua He, Jie Zhang, Man Zhou, and Liu Liu. PanDiT: A few-step diffusion transformer for high-fidelity and efficient pansharpening. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–13, 2025. 3, 11, 12, 14, 16, 18, 20

[13] Andrea Garzelli and Filippo Nencini. Hypercomplex quality assessment of multi/hyperspectral images. *IEEE Geoscience and Remote Sensing Letters*, 6(4):662–665, 2009. 6, 10

[14] 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. 3

[15] Xuanhua He, Keyu Yan, Jie Zhang, Rui Li, Chengjun Xie, Man Zhou, and Danfeng Hong. Multiscale dual-domain guidance network for pan-sharpening. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–13, 2023. 11, 12, 14, 16, 18, 20

[16] Arthur E. Hoerl and Robert W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970. 2

[17] Junming Hou, Xiaoyu Chen, Ran Ran, Xiaofeng Cong, Xinyang Liu, Jian Wei You, and Liang-Jian Deng. Binarized neural network for multi-spectral image fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2236–2245, 2025. 3, 5, 7, 11, 12, 14, 16, 18, 20

[18] 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*, 36(1):1113–1121, 2022. 3, 5, 7, 11, 12, 14, 16, 18, 20, 27

[19] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1646–1654, 2016. 2, 3

[20] Sungpyo Kim, Jeonghyeok Do, Jaehyup Lee, and Munchurl Kim. U-Know-DiffPAN: An uncertainty-aware knowledge distillation diffusion framework with details enhancement for PAN-sharpening. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23069–23079, 2025. 2, 3, 11, 12, 14, 16, 18, 20

[21] Zhuomao Li, Ying Li, Zhihao Li, Yikun Liu, and Gongping Yang. Two-step pansharpening: A high-frequency-guided spatial–spectral enhancement network based on mixture of experts. *IEEE Transactions on Geoscience and Remote Sensing*, 64:1–19, 2026. 3, 5, 7, 11, 12, 14, 16, 18, 20

[22] Xinyang Liu, Junming Hou, Chenxu Wu, Xiaofeng Cong, Zihao Chen, Shangqi Deng, Junling Li, Liang-Jian Deng, and Bo Liu. Physics-informed neural operator for pansharpening. In *Advances in Neural Information Processing Systems*, pages 95622–95652, 2025. 3, 5, 7, 11, 12, 14, 16, 18, 20, 27

[23] Simone Lolli, Luciano Alparone, Andrea Garzelli, and Gemine Vivone. Haze correction for contrast-based multispectral

pansharpening. *IEEE Geoscience and Remote Sensing Letters*, 14(12):2255–2259, 2017. 2, 6, 22, 23

[24] Truong Thanh Nhat Mai, Edmund Y. Lam, and Chul Lee. Deep unfolding tensor rank minimization with generalized detail injection for pansharpening. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–18, 2024. 11, 12, 14, 16, 18, 20

[25] 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. 2, 6, 22, 23

[26] Frosti Palsson, Johannes R. Sveinsson, and Magnus O. Ulfarsson. A new pansharpening algorithm based on total variation. *IEEE Geoscience and Remote Sensing Letters*, 11(1): 318–322, 2014. 3, 6, 22, 23

[27] Frosti Palsson, Johannes R. Sveinsson, Magnus Orn Ulfarsson, and Jon Atli Benediktsson. Model-based fusion of multi- and hyperspectral images using PCA and wavelets. *IEEE Transactions on Geoscience and Remote Sensing*, 53 (5):2652–2663, 2015. 3, 6, 22, 23

[28] 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*, 62:1–16, 2024. 2, 3

[29] Rocco Restaino, Gemine Vivone, Mauro Dalla Mura, and Jocelyn Chanussot. Fusion of multispectral and panchromatic images based on morphological operators. *IEEE Transactions on Image Processing*, 25(6):2882–2895, 2016. 2, 6, 22, 23

[30] Jiangtong Tan, Jie Huang, Naishan Zheng, Man Zhou, Keyu Yan, Danfeng Hong, and Feng Zhao. Revisiting spatial-frequency information integration from a hierarchical perspective for panchromatic and multi-spectral image fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 25922–25931, 2024. 3, 5, 7, 11, 12, 14, 16, 18, 20

[31] Gemine Vivone. Robust band-dependent spatial-detail approaches for panchromatic sharpening. *IEEE Transactions on Geoscience and Remote Sensing*, 57(9):6421–6433, 2019. 2, 6, 22, 23

[32] Gemine Vivone, Luciano Alparone, Jocelyn Chanussot, Mauro Dalla Mura, Andrea Garzelli, Giorgio A. Licciardi, Rocco Restaino, and Lucien Wald. A critical comparison among pansharpening algorithms. *IEEE Transactions on Geoscience and Remote Sensing*, 53(5):2565–2586, 2015. 2, 10

[33] Gemine Vivone, Miguel Simões, Mauro Dalla Mura, Rocco Restaino, José M. Bioucas-Dias, Giorgio A. Licciardi, and Jocelyn Chanussot. Pansharpening based on semiblind deconvolution. *IEEE Transactions on Geoscience and Remote Sensing*, 53(4):1997–2010, 2015. 3, 6, 22, 23

[34] 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. 2, 6, 22, 23

[35] Gemine Vivone, Mauro Dalla Mura, Andrea Garzelli, and Fabio Pacifici. A benchmarking protocol for pansharpening: Dataset, preprocessing, and quality assessment. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:6102–6118, 2021. 10

[36] Gemine Vivone, Mauro Dalla Mura, Andrea Garzelli, Rocco Restaino, Giuseppe Scarpa, Magnus O. Ulfarsson, Luciano Alparone, and Jocelyn Chanussot. A new benchmark based on recent advances in multispectral pansharpening: Revisiting pansharpening with classical and emerging pansharpening methods. *IEEE Geoscience and Remote Sensing Magazine*, 9(1):53–81, 2021. 2, 6, 10

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

[38] Yingying Wang, Xuanhua He, Chen Wu, Jialing Huang, Suiyun Zhang, Rui Liu, Xinghao Ding, and Haoxuan Che. MMMamba: A versatile cross-modal in context fusion framework for pan-sharpening and zero-shot image enhancement. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(22):18656–18664, 2026. 11, 12, 14, 16, 18, 20

[39] Zhou Wang, Alan Conrad Bovik, Hamid Rahim Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 6, 10

[40] Zhiyuan Wang, Dong Li, Kaixin Fu, Chunhui Luo, and Xueyang Fu. Complex-valued residual diffusion with GRPO for pansharpening. In *Proceedings of the Thirty-Fifth International Joint Conference on Artificial Intelligence*, pages 1803–1811. International Joint Conferences on Artificial Intelligence Organization, 2026. 3

[41] 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, 6, 7, 22, 23, 27

[42] Yinghui Xing, Litao Qu, Shizhou Zhang, Jiapeng Feng, Xiuwei Zhang, and Yanning Zhang. Empower generalizability for pansharpening through text-modulated diffusion model. *IEEE Transactions on Geoscience and Remote Sensing*, 62: 1–12, 2024. 3

[43] Gang Yang, Man Zhou, Keyu Yan, Aiping Liu, Xueyang Fu, and Fan Wang. Memory-augmented deep conditional unfolding network for pan-sharpening. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1788–1797, 2022. 3, 5, 7, 11, 12, 14, 16, 18, 20

[44] Gang Yang, Xiangyong Cao, Wenzhe Xiao, Man Zhou, Aiping Liu, Xun Chen, and Deyu Meng. PanFlowNet: A flow-based deep network for pan-sharpening. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16857–16867, 2023. 3, 5, 7, 11, 12, 14, 16, 18, 20

[45] Junfeng Yang, Xueyang Fu, Yuwen Hu, Yue Huang, Xinghao Ding, and John Paisley. PanNet: A deep network architecture for pan-sharpening. In *Proceedings of the IEEE Inter-*

national Conference on Computer Vision, pages 5449–5457, 2017. 2, 3, 5, 7, 11, 12, 14, 16, 18, 20, 27

[46] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017. 2, 3

[47] Man Zhou, Keyu Yan, Jie Huang, Zihe Yang, Xueyang Fu, and Feng Zhao. Mutual information-driven pan-sharpening. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1798–1808, 2022. 3, 5, 7, 11, 12, 14, 16, 18, 20

[48] Zeyu Zhu, Xiangyong Cao, Man Zhou, Junhao Huang, and Deyu Meng. Probability-based global cross-modal upsampling for pansharpening. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14039–14048, 2023. 3