When Latents Forget Pixels: Restoring Fidelity in Diffusion Transformer Super-Resolution

Yu Shi Dartmouth College
Yuyao Zhang Dartmouth College
Yu-Wing Tai Dartmouth College

Project: `https://rain-sy.github.io/projects/PGSR/`



Figure 1: Super-resolution results using **PGSR** compared to one of the other State-of-the-art pixel-guided methods. While it produces visually plausible textures, it hallucinates structures inconsistent with the input (e.g., incorrect texts or unexpectedly sharpened edges). Our method preserves LR pixel features before VAE compression and uses them to guide latent trajectory and ground final rendering, producing sharper results that remain faithful to the LR observation.

Abstract

Image super-resolution (SR) with large generative models has recently achieved remarkable perceptual quality, yet maintaining fidelity to the LR observation remains challenging. In particular, we observe that diffusion transformers (DiTs) built on latent representations suffer from a critical limitation: the compression bottleneck of the VAE weakens fine-grained spatial information, leading to hallucinated details that are weakly grounded in the input image. In this work, we revisit generative SR from a representation perspective and propose a **pixel-grounded super-resolution (PGSR)** framework that preserves LR-observed pixel evidence before VAE compression and reuses it throughout restoration. Instead of relying solely on the compressed latent condition, PGSR extracts pre-VAE pixel evidence from the upsampled LR image and reuses it at two stages. First, **Condition-Side Trajectory Guidance** fuses LR-derived pixel evidence with the latent LR condition to guide the latent restoration trajectory. Second, **Decoder-Side Pixel Grounding** injects multi-scale pixel features into the frozen VAE decoder to ground the final rendering with LR-observed cues. To efficiently adapt large pretrained DiT models, we keep the latent autoencoder and main flow-matching backbone frozen, and train only lightweight restoration modules. We further study an efficient local-window attention variant for improved high-resolution efficiency and scalability. Extensive experiments demonstrate that PGSR improves the realism–fidelity trade-off and produces more faithful, visually convincing results than existing latent generative SR approaches. Code is available at `https://github.com/Rain-sy/PGSR`.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

1 Introduction

Image super-resolution (SR) aims to recover high-frequency details from a low-resolution (LR) input while preserving fidelity to the observed image. Recent approaches leverage large pretrained generative models, including diffusion and transformer-based architectures, to synthesize realistic textures that exceed the capabilities of regression- or GAN-based methods [Ledig et al., 2017, Wang et al., 2018, Saharia et al., 2021, Wang et al., 2023b, Wu et al., 2023]. However, despite their strong perceptual quality, these methods still struggle to remain consistent with the input, often hallucinating details that are only weakly supported by the LR image.

In this work, we identify a key but underexplored reason for this failure: *latent generative models operate on compressed representations that can weaken fidelity-critical pixel evidence*. Modern diffusion transformers and latent diffusion models rely on a VAE tokenizer [Kingma and Welling, 2014, Rombach et al., 2022] to map images into a compact latent space. While this enables efficient high-resolution generation, it also introduces an information bottleneck that removes subtle spatial cues and high-frequency structures essential for faithful SR. As illustrated in Fig. 1, even when conditioned on the LR input, latent-only generation tends to produce visually convincing but incorrect details, revealing a fundamental mismatch between latent generative priors and pixel-level restoration objectives.

This perspective suggests that improving SR is not merely a matter of stronger conditioning, but of *keeping LR-observed image evidence accessible before it is compressed into the latent space.* Prior works have attempted to mitigate this issue by introducing pixel-aware guidance or additional control mechanisms within diffusion-based SR pipelines [Wu et al., 2023, Yang et al., 2024, Sun et al., 2025, Arora et al., 2025]. While effective, most of these designs still use pixel evidence primarily as an external condition for the latent denoiser. They therefore leave two fidelity gaps only partially addressed: the latent guided trajectory may drift away from LR-observed structure, and the frozen decoder may further lose spatial detail during final rendering.

To this end, we propose a **pixel-grounded super-resolution (PGSR)** framework that explicitly preserves LR-observed pixel cues before VAE compression and reuses them throughout restoration. We refer to this design as Pixel-Grounded Guidance: pixel evidence denotes image-domain features extracted from the LR input before VAE compression and used to restore fidelity-critical spatial information. PGSR realizes this idea through two complementary mechanisms. First, Condition-Side Trajectory Guidance fuses LR-derived pixel evidence into the ControlNet condition to anchor the flow trajectory. Second, Decoder-Side Pixel Grounding injects multi-scale pixel features into the frozen VAE decoder to recover fine spatial structures at rendering time. In this way, the pretrained latent backbone provides strong semantic and generative priors, while the pixel-grounded pathway constrains restoration with LR-observed image evidence.

Importantly, our design differs from existing pixel-guided SR methods in that pixel evidence is not only appended as an auxiliary condition. Instead, it is used to ground both what content is generated in latent space and how that content is decoded back into pixels. By explicitly modeling the residual information missing from the latent space, our approach directly targets the root cause of fidelity degradation in generative SR.

To enable practical deployment on large pretrained models, we adopt a parameter-efficient adaptation strategy based on a ControlNet-style architecture [Zhang et al., 2023], where the diffusion transformer backbone remains frozen and only lightweight modules are trained. We further adapt the frozen flow-matching transformer with low-rank adapters and local-window attention to improve high-resolution efficiency.

In summary, our contributions are three-fold:

- We identify latent compression as a fundamental bottleneck in generative super-resolution, and reinterpret SR as a mismatch between latent generative priors and pixel-faithful restoration.
- We propose Pixel-Grounded Guidance, a dual-stage design that reuses pre-VAE pixel evidence to guide both latent trajectory formation and final image decoding, improving fidelity throughout the restoration process.
- We present a practical and efficient implementation on pretrained latent flow-matching models, using lightweight adaptation and local attention to achieve a better realism–fidelity trade-off.

2 Related Work

Generative Super-Resolution and the Realism–Fidelity Trade-off. Early super-resolution (SR) methods Chang et al. [2004], Tai et al. [2006], Sun et al. [2008], Tai et al. [2010] focused on distortion minimization, optimizing metrics such as PSNR and SSIM [Wang et al., 2004], with representative models including SRCNN [Dong et al., 2014], VDSR [Kim et al., 2016], and transformer-based restorers such as SwinIR [Liang et al., 2021]. While effective on distortion metrics, these approaches often produce over-smoothed results.

Perceptual SR methods emphasize visual realism instead. SRGAN [Ledig et al., 2017] and ESRGAN [Wang et al., 2018] show that adversarial training and perceptual losses [Johnson et al., 2016, Zhang et al., 2018] yield sharper, more realistic textures. This shift is formalized by the perception–distortion trade-off [Blau and Michaeli, 2018], where improved perceptual quality often reduces fidelity. GLEAN [Chan et al., 2021] further leverages pretrained generative priors for large-scale SR.

For real-world SR, degradation modeling is critical. BSRGAN and Real-ESRGAN [Zhang et al., 2021, Wang et al., 2021] highlight the importance of realistic blur–noise–compression pipelines. Diffusion-based SR improves perceptual quality and stability: SR3 [Saharia et al., 2021] and DDRM [Kawar et al., 2022] introduce diffusion priors, while ResShift [Yue et al., 2023], SinSR [Wang et al., 2023c], and OSEDiff [Wu et al., 2024] improve efficiency. Recent work explores rectified-flow formulations such as FlowSR and FluxSR [Xu et al., 2025, Li et al., 2025]. Another direction adapts pretrained generative priors, such as StableSR [Wang et al., 2023b], DiffBIR [Lin et al., 2024], XPSR [Qu et al., 2024], and SeeSR [Wu et al., 2023], to enhance restoration via semantic and structural conditioning.

Despite these advances, the realism–fidelity trade-off persists: stronger generative priors improve perceptual quality by relaxing constraints from the LR input, often introducing hallucinated details.

Limitations of Latent-Space Diffusion for Restoration. Latent diffusion models (LDMs) [Rombach et al., 2022] enable scalable high-resolution generation by operating on compressed representations, but this efficiency introduces an information bottleneck that discards fine-grained spatial details critical for faithful restoration. Recent analyses highlight a reconstruction–generation trade-off, where stronger compression benefits synthesis but degrades input fidelity [Yao et al., 2025].

This issue is especially pronounced in SR, where subtle pixel-level structures must be preserved. Prior works mitigate it by strengthening conditioning or modifying inference [Wang et al., 2023b, Lin et al., 2024, Wang et al., 2023c], yet the underlying loss of pixel-domain information remains unresolved. Recent studies therefore revisit pixel-space modeling. Simple Diffusion [Hoogeboom et al., 2023] and Hourglass Diffusion Transformers [Crowson et al., 2024] show that pixel-space generation can scale with proper design, while JiT [Li and He, 2025] and Latent Forcing [Baade et al., 2026] further question the need for aggressive compression.

These findings suggest that preserving or reintroducing pixel-space information is key to faithful reconstruction. Rather than discarding latent priors, our approach compensates for their information loss by reusing pre-VAE pixel evidence throughout restoration.

Pixel-Guided and Grounded Super-Resolution. To mitigate fidelity loss, recent SR methods incorporate stronger conditioning into pretrained generative models. ControlNet [Zhang et al., 2023] enables spatially aligned conditioning and is widely used in restoration pipelines such as DiffBIR [Lin et al., 2024]. Adapter-based methods, including T2I-Adapter [Mou et al., 2024] and LoRA [Hu et al., 2021], provide parameter-efficient task-specific guidance.

More explicitly, pixel-aware SR methods aim to preserve local structures. PASD [Yang et al., 2024] introduces pixel-level guidance into latent diffusion. SeeSR [Wu et al., 2023], FaithDiff [Chen et al., 2025], and GenDR [Wang et al., 2025] improve fidelity via semantic and structural conditioning. PiSA-SR [Sun et al., 2025] disentangles pixel fidelity and semantic enhancement with dual LoRA modules, while GuideSR [Arora et al., 2025] adds a full-resolution guidance branch. However, these methods typically treat pixel information as auxiliary to latent generation, leaving two issues: (1) latent trajectories may drift from LR structure, and (2) VAE decoding can further lose spatial detail.

Our proposed Pixel-Grounded Super-Resolution (PGSR) addresses both issues by reusing pre-VAE pixel evidence at two stages: guiding the latent restoration trajectory and grounding the final VAE decoding. This dual design directly targets the information bottleneck of latent generative SR, enabling more faithful and consistent reconstruction.

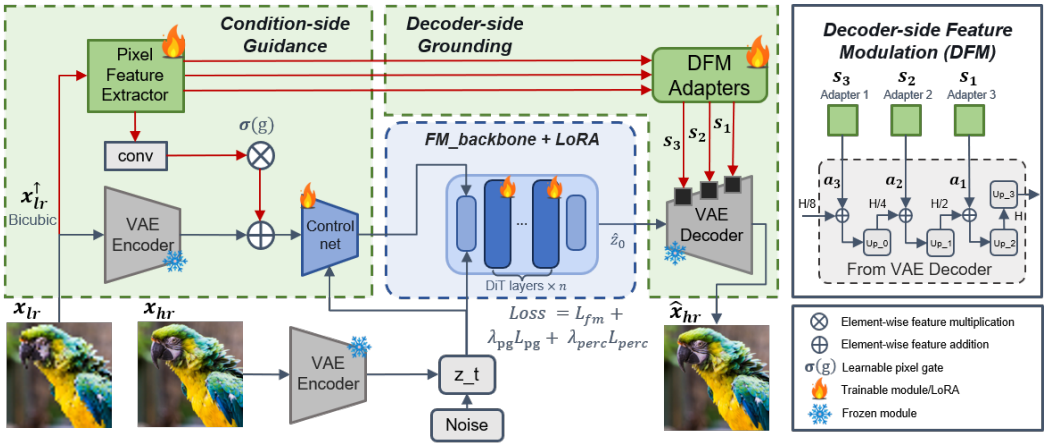


Figure 2: Overview of the training pipeline of PGSR. The LR observation is first aligned to the HR grid and encoded into a latent condition. PGSR introduces pixel-grounded evidence at two complementary locations: Condition-Side Trajectory Guidance fuses LR-derived pixel evidence with the latent condition to anchor the flow-matching trajectory, while Decoder-Side Pixel Grounding injects multi-scale pixel features into the frozen VAE decoder for faithful rendering.

3 Methodology

In this section, we present PGSR as a pixel-grounded latent flow-matching framework for super-resolution. Instead of using the LR image only after VAE encoding, PGSR leverages pre-VAE pixel evidence to guide both latent trajectory formation and final rendering, addressing two fidelity gaps: VAE compression weakens LR structure, and the frozen decoder may lose spatial detail.

As shown in Fig. 2, PGSR reuses pre-VAE pixel evidence at two stages via Pixel-Grounded Guidance: condition-side trajectory guidance anchors the latent flow to LR structure, while decoder-side grounding injects multi-scale cues into the frozen VAE decoder for faithful reconstruction.

We first formulate latent flow-matching SR, then introduce shared pixel evidence with condition-side guidance and decoder-side grounding, followed by the learning objective and a parameter-efficient design.

3.1 Preliminaries on Latent Flow-Matching Super-Resolution

Let $x_{hr} \in \mathbb{R}^{H\times W\times 3}$ be a high-resolution image and $x_{lr} \in \mathbb{R}^{\frac{H}{s}\times\frac{W}{s}\times 3}$ its low-resolution observation under degradation $\mathcal{D}$ with scale s, i.e., $x_{lr} = \mathcal{D}(x_{hr})$.

We upsample x_{lr} to the HR grid, $x_{lr}^{\uparrow} = \mathrm{Bicubic}(x_{lr})$, and encode both images with a frozen VAE:

$$z_{lr} = E_{\mathrm{vae}}(x_{lr}^{\uparrow}), \quad z_{hr} = E_{\mathrm{vae}}(x_{hr}).$$

We learn a conditional velocity predictor in latent space along the flow-matching trajectory. Given $\epsilon \sim \mathcal{N}(0, I)$ and noise level σ_t,

$$z_t = \sigma_t\epsilon + (1-\sigma_t)z_{hr}, \quad v^\star = \epsilon - z_{hr}.$$

The model predicts $\hat{v}$ conditioned on the LR input and estimates

$$\hat{z}_0 = z_t - \sigma_t\hat{v}, \quad \mathcal{L}_{\mathrm{fm}} = \|\hat{v} - v^\star\|_2^2.$$

While efficient, this formulation conditions on the compressed latent z_{lr}. We therefore introduce pixel grounding to preserve pre-VAE evidence and guide both the latent trajectory and final decoding.

3.2 Pixel-Grounded Super-Resolution

Instead of being a global semantic context, PGSR keeps the LR observation available outside the VAE bottleneck by extracting pixel evidence *before latent encoding*.

Pre-VAE pixel evidence. Given the upsampled LR image $x_{lr}^{\uparrow}$, a lightweight pixel encoder $P(\cdot)$ produces a latent-resolution feature and multi-scale decoder features:

$$\big(\bar{p}_{lr}, \mathcal{S}_{lr}\big) = P(x_{lr}^{\uparrow}), \qquad \mathcal{S}_{lr} = \{s_1, s_2, s_3\}.$$

Here $\bar{p}_{lr}$ is used to ground the latent trajectory, while $\{s_1, s_2, s_3\}$ are used to ground the final VAE decoding process.

This pixel evidence does not provide unobserved HR details; instead, it preserves LR-observed cues such as edges, boundaries, and local color transitions before they are compressed by the VAE.

Condition-Side Trajectory Guidance. The latent LR condition z_{lr} provides a compact structural anchor, while $\bar{p}_{lr}$ supplies complementary pre-VAE image evidence.

We fuse them through a gated residual correction: $\tilde{z}_{lr} = z_{lr} + \lambda\, \sigma(g)\, W_p(\bar{p}_{lr})$, where W_p is a 1×1 projection, g is a learnable scalar gate, $\sigma(\cdot)$ is the sigmoid function, and λ controls the pixel-conditioning strength. The grounded condition is then packed and passed to the ControlNet branch:

$$\Delta_t = C_\phi(\mathrm{Pack}(z_t), \mathrm{Pack}(\tilde{z}_{lr}), t), \qquad \hat{v} = F_{\psi,\omega}(\mathrm{Pack}(z_t), t; \Delta_t),$$

where $F_{\psi,\omega}$ denotes the pretrained flow-matching transformer with frozen weights ψ and trainable LoRA parameters ω. Here $\mathrm{Pack}(\cdot)$ denotes the standard rearrangement from latent feature maps to the image-token sequence consumed by the transformer. This grounds the latent trajectory without replacing the compact latent condition.

Decoder-Side Pixel Grounding. Condition-Side Trajectory Guidance constrains the predicted latent, but the final image is still rendered by the frozen VAE decoder. To ground this decoding step, PGSR injects the multi-scale pixel features into matched decoder stages through decoder-side feature modulation (DFM). Let f_d^{ℓ} denote the decoder activation before upsampling block ℓ. We apply

$$\tilde{f}_d^{\ell} = f_d^{\ell} + Z_\ell(\mathrm{SiLU}(A_\ell(s_\ell))), \qquad \ell \in \{0, 1, 2\},$$

where s_ℓ denotes the scale-matched pixel tap (s_3 for $\ell = 0$ at $H/8$, s_2 for $\ell = 1$ at $H/4$, and s_1 for $\ell = 2$ at $H/2$), A_ℓ aligns the pixel feature to the decoder width, and Z_ℓ is a zero-initialized output projection. The final prediction is obtained by $\hat{x}_{hr} = D_{\mathrm{vae}}^{\mathrm{pg}}(\hat{z}_0; \mathcal{S}_{lr})$. Here $D_{\mathrm{vae}}^{\mathrm{pg}}$ denotes the frozen VAE decoder equipped with DFM pixel-grounding adapters.

Because Z_ℓ is zero-initialized, the decoder is identical to the frozen VAE at initialization and DFM gradually learns pixel-grounded rendering during training.

3.3 Efficient Architecture Adaptation

PGSR is a parameter-efficient adaptation of a pretrained latent flow-matching DiT. We freeze the latent autoencoder and generative backbone, and train only the ControlNet restoration branch, LoRA adapters, pre-VAE pixel encoder, gated trajectory-guidance fusion, and decoder-side pixel-grounding adapters. As SR is purely image-conditioned, we use cached empty-text embeddings and drive restoration solely from the LR input.

For high-resolution efficiency, we adopt local-window attention by replacing dense image-token attention in selected blocks with pretrained local processors:

$$\mathrm{Attn}(q_i) = \sum_{j \in \mathcal{N}(i) \cup \mathcal{G}} \alpha_{ij} v_j,$$

where $\mathcal{N}(i)$ is a local neighborhood and $\mathcal{G}$ denotes a set of optional global tokens. PGSR is fine-tuned on this accelerated backbone with the same grounding pathway and objective.

The training objective combines latent flow supervision with pixel-grounded fidelity:

$$\mathcal{L} = \mathcal{L}_{\mathrm{fm}} + \lambda_{\mathrm{perc}}\mathcal{L}_{\mathrm{perc}} + \lambda_{\mathrm{pg}}\mathcal{L}_{\mathrm{pg}},$$

where $\mathcal{L}_{\mathrm{fm}}$ is the flow-matching loss. Image-space terms are computed from $\hat{x}_{hr}$:

$$\mathcal{L}_{\mathrm{perc}} = m_{\mathrm{perc}}\, \mathcal{L}_{\mathrm{LPIPS}}(\hat{x}_{hr}, x_{hr}), \quad \mathcal{L}_{\mathrm{pg}} = \frac{1}{\sum_i m_i} \sum_i m_i \|\hat{x}_{hr}^{(i)} - x_{hr}^{(i)}\|_1.$$

Here $m_i = \mathbf{1}[\sigma_i \le \tau]$ is a low-noise mask, with m_{perc} as its perceptual gate. Image-space supervision is applied only at low noise, where $\hat{z}_0 = z_t - \sigma_t \hat{v}$ is stable. Thus, $\mathcal{L}_{\mathrm{fm}}$ guides the full trajectory, while $\mathcal{L}_{\mathrm{perc}}$ and $\mathcal{L}_{\mathrm{pg}}$ refine reliable pixel-grounded outputs.



GT SD_VAE FLUX_VAE

Method	RMSE↓	PSNR↑	SSIM↑
GT	0.00	∞	1.0000
SD VAE	14.42	24.95	0.8454
FLUX VAE	13.77	25.35	0.8528

Figure 3: VAE reconstruction stress test on an example. Left: visual comparison under controlled latent reconstruction. Right: quantitative reconstruction metrics. Both Stable Diffusion VAE and FLUX VAE exhibit substantial information loss due to latent compression, leading not only to blurring but also to severe structural degradation, where characters, digits, and fine patterns become largely unreadable or distorted beyond recognition.

4 Experiments

Datasets. We use two groups of training datasets and three evaluation benchmarks. For clean paired pretraining, we use DF2K, which combines DIV2K [Agustsson and Timofte, 2017] and Flickr2K [Wang et al., 2018], with paired bicubic $\times 4$ LR–HR images. For real-degradation adaptation, we construct a larger mixed HR corpus from the union of DF2K, LSDIR [Li et al., 2023], FFHQ [Karras et al., 2019], and OST [Wang et al., 2018]; LR inputs are synthesized online with a real-world degradation pipeline. For evaluation, we report results on DIV2K validation [Agustsson and Timofte, 2017], RealSR [Cai et al., 2019], and DRealSR [Wei et al., 2020], covering both synthetic bicubic degradation and real captured LR–HR pairs.

Training and inference. We instantiate PGSR with FLUX.1-dev [Black Forest Labs, 2024] and initialize the restoration control branch from a pretrained SR ControlNet compatible with the base flow-matching model [Zhang et al., 2023]. Stage 1 learns the clean SR mapping on DF2K and selects the best checkpoint by DIV2K validation PSNR. Stage 2 fine-tunes from this checkpoint on the mixed HR corpus using the second-order Real-ESRGAN degradation process [Wang et al., 2021], switches validation to real-world data, and selects the main model by RealSR validation LPIPS. In this stage, we enable the decoder-side DFM branch and the low-noise image-space losses while continuing to train the condition-side trajectory guidance modules. At inference time, PGSR follows the pretrained flow-matching sampling trajectory conditioned on the LR observation. Given the LR latent z_{lr} and Gaussian noise ϵ, the initial latent is formed as $z_{t_0}^{\text{init}} = \sigma_{t_0}\epsilon + (1 - \sigma_{t_0})z_{lr}$, where t_0 controls how strongly the sampler departs from the LR latent condition. Unless otherwise specified, we use the full restoration trajectory and decode the final latent with the DFM-enabled VAE decoder.

Evaluation metrics. We evaluate distortion fidelity with PSNR and SSIM [Wang et al., 2004], reference-based perceptual quality with LPIPS [Zhang et al., 2018] and DISTS [Ding et al., 2022], and distribution-level fidelity with FID [Heusel et al., 2017]. We further report no-reference perceptual metrics, including NIQE [Mittal et al., 2013], MUSIQ [Ke et al., 2021], MANIQA [Yang et al., 2022], and CLIP-IQA [Wang et al., 2023a]. Higher is better for PSNR, SSIM, MUSIQ, MANIQA, and CLIP-IQA; lower is better for LPIPS, DISTS, FID, and NIQE.

Implementation details. All experiments are conducted on 8 NVIDIA RTX 6000 Ada GPUs, each with 48GB of memory. Training uses bf16 mixed precision and distributed data parallelism via Hugging Face Accelerate. The VAE and pretrained FLUX backbone are frozen; the trainable components are the ControlNet branch, the shared pixel feature extractor, the condition-side trajectory guidance fusion layer, the decoder-side DFM adapters, and FLUX LoRA adapters. We optimize with AdamW using cosine learning-rate decay and linear warmup. Additional training, adaptation, degradation, and inference details are provided in Appendix A.

Necessity of pixel-level information for decoder-side grounding. We conduct a controlled VAE reconstruction experiment to analyze the fidelity of latent compression in standard encoder-decoder pipelines. Specifically, we feed ground-truth HR images into the VAE and reconstruct them via the standard encode-decode process, using both the Stable Diffusion VAE [Rombach et al., 2022] and FLUX VAE [Black Forest Labs, 2024]. As shown in Fig. 3, both models exhibit noticeable reconstruction degradation, including distorted patterns and blurred fine-grained details, indicating

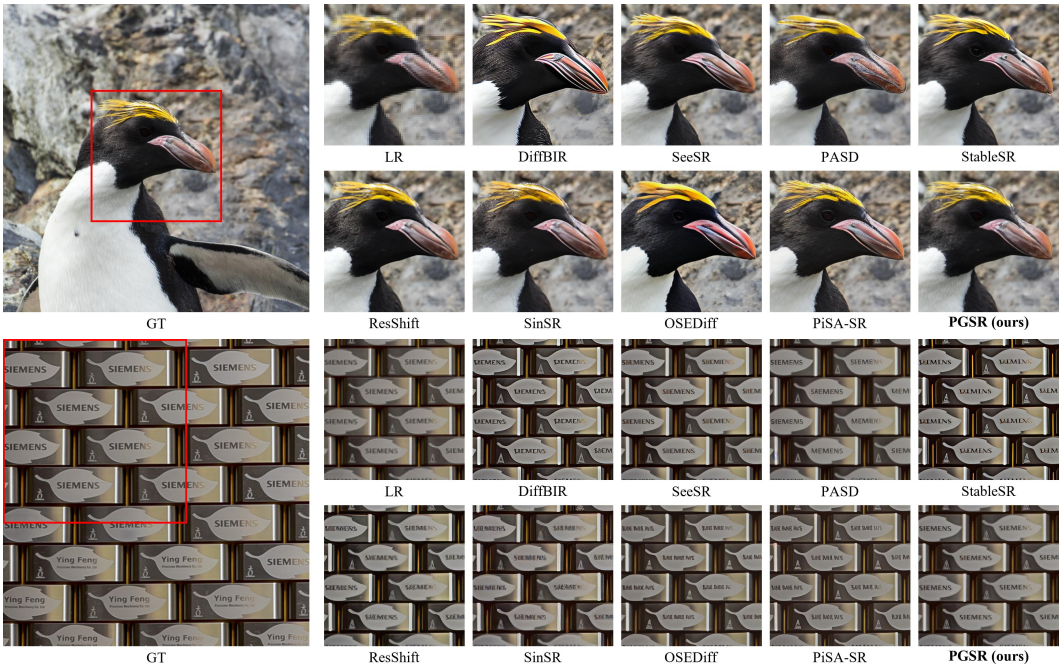


Figure 4: Qualitative comparison on representative $\times 4$ SR examples from both synthetic bicubic degradation and real degradation. PGSR aims to preserve LR-consistent structures while recovering perceptually plausible high-frequency details, including fine feather structures and text characters.

that the VAE bottleneck is inherently lossy even under perfect input conditions. This suggests that encoder-decoder-based SR pipelines do not fully preserve or utilize pixel-level signals during reconstruction, resulting in unavoidable degradation of high-frequency content. These results highlight an important limitation of latent compression in existing generative SR systems: important spatial details are partially discarded during encoding, and cannot be fully recovered by the decoder alone.

4.1 Comparison with State-of-the-Art Methods

Compared methods. We compare PGSR with representative generative SR methods which are reproducible, including ResShift [Yue et al., 2023], StableSR [Wang et al., 2023b], DiffBIR [Lin et al., 2024], SeeSR [Wu et al., 2023], PASD [Yang et al., 2024], OSEDiff [Wu et al., 2024], SinSR [Wang et al., 2023c], and PiSA-SR [Sun et al., 2025].

Qualitative comparisons. Figure 4 shows representative visual comparisons on both synthetic bicubic and real-degradation benchmarks. Compared with prior generative SR methods, PGSR is designed to preserve LR-consistent structures while recovering plausible high-frequency details, especially around repeated textures and small semantic regions. The first row presents a penguin image from the DIV2K-Val dataset with virtual bicubic degradation, where the competing methods either produce overly sharp hair and hallucinated eye details or change the saturation of the image. In contrast, PGSR better preserves the spatial layout inherited from the LR input while producing sharp local details, leading to a more consistent result with the ground-truth. The second row shows a RealSR captured image, where the degradation is real-world degradation rather than a simple bicubic downsampling process. Baselines such as StableSR and DiffBIR hallucinate textures or distort small structures, whereas PGSR preserves object boundaries and correctly renders the text.

Quantitative comparisons. Table 1 reports quantitative comparisons on synthetic and real-world $\times 4$ SR benchmarks using distortion, perceptual, distribution-level, and no-reference quality metrics. On DIV2K-Val, PGSR achieves the best PSNR, LPIPS, DISTS, FID, and NIQE, and ranks second in SSIM, indicating that preserving pre-VAE pixel evidence improves fidelity to the LR observation while maintaining strong perceptual quality. Notably, compared with perception-oriented methods such as DiffBIR and PiSA-SR, PGSR obtains substantially better distortion and perceptual metrics, suggesting

Table 1: Quantitative comparison with representative generative SR methods on synthetic and real-world $\times 4$ benchmarks. All methods are evaluated using the same local metric pipeline, covering distortion metrics, full-reference perceptual metrics, distribution-level fidelity, and no-reference perceptual quality. For each dataset and metric, the best, second-best, and third-best results are highlighted in **bold**, underline, and *italics*, respectively.

Dataset	Method	PSNR↑	SSIM↑	LPIPS↓	DISTS↓	FID↓	NIQE↓	MUSIQ↑	MANIQA↑	CLIP-IQA↑
DIV2K	ResShift	*25.56*	*0.7625*	0.2166	0.0797	*28.36*	4.8223	62.74	0.3555	0.6276
	StableSR	23.55	0.7050	0.2272	0.0850	32.50	3.7174	*68.55*	0.4591	*0.7080*
	DiffBIR	23.04	0.6852	0.2731	0.1054	36.48	3.6071	68.06	**0.4639**	**0.7370**
	SeeSR	25.51	0.7566	0.2225	0.0866	31.19	3.8600	67.11	0.4340	0.6403
	PASD	24.86	0.7373	0.2361	*0.0849*	29.94	*3.3494*	66.66	0.3975	0.5996
	OSEDiff	23.16	0.7058	0.2386	0.1031	34.30	3.4451	69.34	0.4404	0.7004
	SinSR	25.78	**0.7781**	0.2107	0.0862	27.39	4.2933	64.58	0.3881	0.6582
	PiSA-SR	23.59	0.7087	*0.2158*	0.0937	32.71	3.2426	**70.46**	*0.4556*	0.7190
	PGSR (ours)	**25.89**	0.7707	**0.2104**	**0.0792**	**24.88**	**3.2244**	67.93	0.3913	0.6479
DRealSR	ResShift	24.73	0.6702	0.4298	0.1078	54.11	6.6914	31.91	0.3060	0.6069
	StableSR	24.77	0.6934	0.4194	0.1116	55.07	4.5860	*34.78*	0.4094	0.6720
	DiffBIR	24.83	0.5979	0.4582	0.1255	54.30	4.6915	34.75	*0.4132*	0.6890
	SeeSR	25.91	**0.7755**	**0.2761**	0.0962	*46.91*	5.3388	33.33	0.3606	0.5459
	PASD	*25.84*	0.7464	0.2898	0.1087	50.28	6.2480	29.54	0.3194	0.4442
	OSEDiff	24.69	0.7358	0.3244	0.1117	46.61	4.1527	**37.01**	**0.4764**	0.7016
	SinSR	25.47	0.6632	0.4496	*0.1047*	60.62	5.7972	31.33	0.3590	*0.6937*
	PiSA-SR	25.37	0.7592	0.3133	0.1142	49.88	*4.4010*	34.67	0.4456	**0.7171**
	PGSR (ours)	**25.97**	*0.7550*	*0.3054*	**0.0911**	**46.15**	**3.8707**	35.83	0.3144	0.5932
RealSR	ResShift	24.06	0.7112	0.3491	0.1758	63.61	6.9271	55.20	0.3336	0.5647
	StableSR	22.77	0.6974	0.3288	0.1748	63.08	5.0611	62.57	0.4446	0.6284
	DiffBIR	23.74	0.6353	0.3456	0.1807	59.47	4.9181	62.51	*0.4700*	**0.6958**
	SeeSR	25.13	**0.7620**	0.2750	*0.1553*	61.99	5.3505	61.20	0.4380	0.5984
	PASD	**25.20**	*0.7477*	*0.2737*	0.1473	**52.85**	4.8421	58.48	0.3741	0.5182
	OSEDiff	23.68	0.7163	0.2999	0.1609	*58.80*	4.3153	67.29	**0.4758**	0.6832
	SinSR	24.72	0.7080	0.3642	0.1766	67.05	5.8306	57.86	0.3844	0.6405
	PiSA-SR	24.03	0.7307	0.2719	0.1564	61.42	*4.3685*	**67.97**	0.4746	*0.6745*
	PGSR (ours)	*24.75*	0.7495	**0.2596**	**0.1365**	54.63	**4.0470**	*63.39*	0.4116	0.5593

Table 2: Architecture ablation of pixel-grounded pathways in PGSR. The left table reports quantitative results on DIV2K-Val and RealSR, while the right panel shows a representative visual ablation. Removing either the condition-side trajectory guidance or the decoder-side grounding degrades reconstruction and perceptual quality, and the full design gives the best results on both datasets.

Dataset	Variant	PSNR↑	SSIM↑	LPIPS↓
DIV2K	w/o Pixel-Grounded	23.68	0.7592	0.2431
	w/o Condition-Side	24.66	0.7644	0.2204
	w/o Decoder-Side	24.78	0.7651	0.2188
	PGSR (full)	**25.89**	**0.7707**	**0.2104**
RealSR	w/o Pixel-Grounded	22.15	0.7305	0.2789
	w/o Condition-Side	24.22	0.7420	0.2631
	w/o Decoder-Side	24.08	0.7401	0.2626
	PGSR (full)	**24.75**	**0.7495**	**0.2596**



that Pixel-Grounded Guidance reduces visually plausible but input-inconsistent hallucinations. On DRealSR, PGSR obtains the best PSNR, DISTS, FID, and NIQE, showing strong robustness to real-world degradations and degradation traces. On RealSR, PGSR achieves the best LPIPS, DISTS, and NIQE, and ranks second in SSIM and FID. These results demonstrate a favorable realism–fidelity trade-off across both synthetic and real settings: the latent flow-matching backbone provides realistic textures, while trajectory guidance and decoder-side grounding constrain the restoration with LR-observed pixel evidence during latent generation and final rendering.

4.2 Ablation Study

Ablation study on PGSR components. Table 2 shows that removing either grounding pathway degrades PSNR, SSIM, and LPIPS, while removing both causes the largest drop. The full model performs best on both datasets, supporting complementary trajectory and decoder grounding. Loss ablations are provided in Appendix B.2.

Impact of Sparse Attention. At fixed output sizes of 1792^2 (DIV2K-Val) and 1536^2 (RealSR), sparse attention reduces inference time by approximately 37% with a modest quality decrease (Table 3, left). This benchmark is separate from Table 1. Sparse attention also enables 2K $\rightarrow$ 8K SR (Fig. 5); Appendix B.5 reports pixel-grounding FLOPs.

Text preservation. Using PARSeq [Bautista and Atienza, 2022], we evaluate 1,339 words (4–8 LR pixels high) in 100 TextOCR-val images [Singh et al., 2021] with bicubic $\times 4$ degradation, fixed GT quadrilateral crops, and lowercase alphanumeric normalization. PGSR achieves 27.48% word accuracy and 57.33% character error rate (CER) (Table 3, right).

Table 3: Left: efficiency–quality comparison of dense and sparse PGSR with 20 steps. Right: small-text OCR; HR calibrates recognizer error. Bold: best restored-input result.

Dataset	Attention	Efficiency		Quality		
		s/img↓	s/step↓	PSNR↑	SSIM↑	LPIPS↓
DIV2K	Full	61.81	3.09	26.36	0.7710	0.2571
	Sparse	38.95	1.95	25.81	0.7679	0.2603
RealSR	Full	58.28	2.91	24.88	0.7501	0.2582
	Sparse	37.09	1.85	24.37	0.7482	0.2601

Method	Acc. (%)↑	CER (%)↓
HR reference	86.41	4.97
PASD	8.36	82.00
SeeSR	7.17	85.77
PiSA-SR	7.32	82.40
PGSR	**27.48**	**57.33**



Figure 5: 8K Super-resolution demo using our method. The super-resolution results demonstrate clear quality improvements over the LR input on the DIV8K dataset.

Limitations. PGSR inherits both the strengths and costs of large pretrained flow-matching models. Due to the large ControlNet and the backbone, training requires substantial memory, and inference remains slower than one-step SR models. Besides, since our losses emphasize LR consistency and a mean-error learning objective, PGSR can produce slightly smoother textures than methods that aggressively optimize perceptual sharpness. This reflects a realism–fidelity trade-off; future work may improve sharpness through distillation while preserving input-output consistency.

5 Conclusion

We presented PGSR, a generative super-resolution framework centered on Pixel-Grounded Guidance. Rather than treating the LR image only as a latent condition, PGSR reuses pre-VAE pixel evidence through two complementary mechanisms: Condition-Side Trajectory Guidance anchors the ControlNet condition and latent flow trajectory, while Decoder-Side Pixel Grounding injects multi-scale pixel evidence into the frozen VAE decoder. This dual design directly addresses the fidelity loss caused by latent compression and provides a practical path toward more faithful generative SR with large pretrained diffusion transformers.

References

Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017.

Aditya Arora, Zhengzhong Tu, Yufei Wang, Ruizheng Bai, Jian Wang, and Sizhuo Ma. Guidesr: Rethinking guidance for one-step high-fidelity diffusion-based super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2025.

Alan Baade, Eric Ryan Chan, Kyle Sargent, Changan Chen, Justin Johnson, Ehsan Adeli, and Li Fei-Fei. Latent forcing: Reordering the diffusion trajectory for pixel-space image generation. *arXiv preprint arXiv:2602.11401*, 2026.

Darwin Bautista and Rowel Atienza. Scene text recognition with permuted autoregressive sequence models. In *European Conference on Computer Vision (ECCV)*, 2022.

Black Forest Labs. Flux. 2024. URL `https://github.com/black-forest-labs/flux`.

Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Jianrui Cai, Hui Zeng, Hongwei Yong, Zisheng Cao, and Lei Zhang. Toward real-world single image super-resolution: A new benchmark and a new model. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

Kelvin C. K. Chan, Xintao Wang, Xiangyu Xu, Jinjin Gu, and Chen Change Loy. Glean: Generative latent bank for large-factor image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Hong Chang, Dit-Yan Yeung, and Yimin Xiong. Super-resolution through neighbor embedding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004.

Junyang Chen, Jinshan Pan, and Jiangxin Dong. Faithdiff: Unleashing diffusion priors for faithful image super-resolution. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 28188–28197, 2025.

Katherine Crowson, Stefan Andreas Baumann, Alex Birch, Tanishq Mathew Abraham, Daniel Z Kaplan, and Enrico Shippole. Scalable high-resolution pixel-space image synthesis with hourglass diffusion transformers. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 9550–9575, 2024.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli. Image quality assessment: Unifying structure and texture similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2567–2581, 2022.

Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *European Conference on Computer Vision (ECCV)*, 2014.

Song Fei, Tian Ye, Lujia Wang, and Lei Zhu. LucidFlux: Caption-free photo-realistic image restoration via a large-scale diffusion transformer. *arXiv preprint arXiv:2509.22414*, 2025.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Emiel Hoogeboom, Jonathan Heek, and Tim Salimans. Simple diffusion: End-to-end diffusion for high resolution images. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 13240–13258, 2023.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision (ECCV)*, 2016.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4401–4410, 2019.

Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. In *Advances in neural information processing systems (NeurIPS)*, 2022.

Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. Musiq: Multi-scale image quality transformer. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5148–5157, 2021.

Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations (ICLR)*, 2014.

Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Jianze Li, Jiezhang Cao, Yong Guo, Wenbo Li, and Yulun Zhang. One diffusion step to real-world super-resolution via flow trajectory distillation. In *International Conference on Machine Learning (ICML)*, pages 34044–34053, 2025.

Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. *arXiv preprint arXiv:2511.13720*, 2025.

Yawei Li, Kai Zhang, Jingyun Liang, Jiezhang Cao, Ce Liu, Rui Gong, Yulun Zhang, Hao Tang, Yun Liu, Denis Demandolx, Rakesh Ranjan, Radu Timofte, and Luc Van Gool. Lsdir: A large scale dataset for image restoration. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1775–1787, 2023.

Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *IEEE/CVF International Conference on Computer Vision Workshops*, 2021.

Xinqi Lin, Jingwen He, Ziyan Chen, Zhaoyang Lyu, Bo Dai, Fanghua Yu, Wanli Ouyang, Yu Qiao, and Chao Dong. Diffbir: Towards blind image restoration with generative diffusion prior. In *European Conference on Computer Vision (ECCV)*, 2024.

Anish Mittal, Rajiv Soundararajan, and Alan C. Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212, 2013.

Chong Mou, Xintao Wang, Liangbin Xie, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.

Yunpeng Qu, Kun Yuan, Kai Zhao, Qizhi Xie, Jinhua Hao, Ming Sun, and Chao Zhou. Xpsr: Cross-modal priors for diffusion-based image super-resolution. In *European Conference on Computer Vision (ECCV)*, 2024.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J. Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *arXiv preprint arXiv:2104.07636*, 2021.

Amanpreet Singh, Guan Pang, Mandy Toh, Jing Huang, Wojciech Galuba, and Tal Hassner. TextOCR: Towards large-scale end-to-end reasoning for arbitrary-shaped scene text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8802–8812, 2021.

Jian Sun, Zongben Xu, and Heung-Yeung Shum. Image super-resolution using gradient profile prior. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

Lingchen Sun, Rongyuan Wu, Zhiyuan Ma, Shuaizheng Liu, Qiaosi Yi, and Lei Zhang. Pixel-level and semantic-level adjustable super-resolution: A dual-lora approach. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Yu-Wing Tai, Wai-Shun Tong, and Chi-Keung Tang. Perceptually-inspired and edge-directed color image super-resolution. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

Yu-Wing Tai, Shuaicheng Liu, Michael S Brown, and Stephen Lin. Super resolution using edge prior and single image detail synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

Jianyi Wang, Kelvin C. K. Chan, and Chen Change Loy. Exploring clip for assessing the look and feel of images. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(2):2555–2563, 2023a.

Jianyi Wang, Zongsheng Yue, Shangchen Zhou, Kelvin C. K. Chan, and Chen Change Loy. Exploiting diffusion prior for real-world image super-resolution. *arXiv preprint arXiv:2305.07015*, 2023b.

Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. *arXiv preprint arXiv:1809.00219*, 2018.

Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. *arXiv preprint arXiv:2107.10833*, 2021.

Yan Wang, Shijie Zhao, Kexin Zhang, Junlin Li, and Li Zhang. Gendr: Lighten generative detail restoration. *arXiv preprint arXiv:2503.06790*, 2025.

Yufei Wang, Wenhan Yang, Xinyuan Chen, Yaohui Wang, Lanqing Guo, Lap-Pui Chau, Ziwei Liu, Yu Qiao, Alex C. Kot, and Bihan Wen. Sinsr: Diffusion-based image super-resolution in a single step. *arXiv preprint arXiv:2311.14760*, 2023c.

Zhou Wang, Alan Bovik, Hamid Sheikh, and Eero Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 2004.

Pengxu Wei, Ziwei Xie, Hannan Lu, Zongyuan Zhan, Qixiang Ye, Wangmeng Zuo, and Liang Lin. Component divide-and-conquer for real-world image super-resolution. In *European Conference on Computer Vision (ECCV)*, 2020.

Rongyuan Wu, Tao Yang, Lingchen Sun, Zhengqiang Zhang, Shuai Li, and Lei Zhang. Seesr: Towards semantics-aware real-world image super-resolution. *arXiv preprint arXiv:2311.16518*, 2023.

Rongyuan Wu, Lingchen Sun, Zhiyuan Ma, and Lei Zhang. One-step effective diffusion network for real-world image super-resolution. In *Neural Information Processing Systems (NeurIPS)*, 2024.

Jiaqi Xu, Wenbo Li, Haoze Sun, Fan Li, Zhixin Wang, Long Peng, Jingjing Ren, Haoran Yang, Xiaowei Hu, Renjing Pei, and Pheng-Ann Heng. Fast image super-resolution via consistency rectified flow. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11755–11765, 2025.

Sidi Yang, Tianhe Wu, Shuwei Shi, Shanshan Lao, Yuan Gong, Mingdeng Cao, Jiahao Wang, and Yujiu Yang. Maniqa: Multi-dimension attention network for no-reference image quality assessment. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1191–1200, 2022.

Tao Yang, Rongyuan Wu, Peiran Ren, Xuansong Xie, and Lei Zhang. Pixel-aware stable diffusion for realistic image super-resolution and personalized stylization. In *European Conference on Computer Vision (ECCV)*, 2024.

Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Zongsheng Yue, Jianyi Wang, and Chen Change Loy. Resshift: Efficient diffusion model for image super-resolution by residual shifting. *arXiv preprint arXiv:2307.12348*, 2023.

Kai Zhang, Jingyun Liang, Luc Van Gool, and Radu Timofte. Designing a practical degradation model for deep blind image super-resolution. *arXiv preprint arXiv:2103.14006*, 2021.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

A Additional Implementation Details

We provide additional implementation details to facilitate reproducibility. Unless otherwise stated, all reported PGSR models are initialized from `black-forest-labs/FLUX.1-dev`. The restoration ControlNet branch is initialized from a FLUX-compatible SR ControlNet checkpoint, while the FLUX backbone and the VAE encoder–decoder remain frozen throughout training. The trainable components include the ControlNet restoration branch, the pre-VAE pixel extractor, the gated condition-side pixel fusion layer, decoder-side DFM adapters, and LoRA adapters inserted into the FLUX backbone.

Optimization. We train PGSR with AdamW using a base learning rate of 1×10^{-5}, weight decay 0.01, and Adam coefficients $(\beta_1, \beta_2) = (0.9, 0.999)$. The learning rate is linearly warmed up for the first 5 epochs and then decayed with a cosine schedule. The ControlNet branch uses the base learning rate, while the pixel branch and LoRA parameters use a higher learning rate of 1×10^{-4}. Training is performed with bfloat16 mixed precision on 8 NVIDIA RTX 6000 Ada GPUs. Unless otherwise specified, we use a per-GPU batch size of 4 and gradient accumulation of 8, resulting in an effective batch size of 256. We use a fixed random seed of 42 for the reported runs.

Training protocol. For the clean paired stage, we train on DF2K with paired bicubic $\times 4$ LR–HR images. HR crops of size 512×512 are used, corresponding to 128×128 LR crops for $\times 4$ SR. We sample two crops per image and train for 60 epochs. The clean-stage checkpoint is selected according to DIV2K validation PSNR. For the real-world degradation stage, we fine-tune the clean checkpoint using a Real-ESRGAN-style second-order degradation process on the mixed HR corpus described in Sec. 4. The real-world stage uses validation on real captured data, and the checkpoint used for the main real-world results is selected according to RealSR validation LPIPS.

Model adaptation. The condition-side pixel feature is fused with the LR latent condition through the gated residual formulation described in Sec. 3.2. The pixel gate is initialized with a positive logit, so that pixel evidence is active at the beginning of training while still being learnable. LoRA adapters use rank 16, alpha 16, zero dropout, and Gaussian initialization. Since SR is fully image-conditioned, we use cached empty-text embeddings during training and inference. In our implementation, we additionally allow a small set of learnable leading text tokens with a substantially smaller learning rate and L2 regularization; this component is used only as a lightweight prior adapter and is not the main source of image conditioning.

Decoder-side pixel grounding. The decoder-side DFM branch is enabled for the full PGSR model. Multi-scale pixel features are injected into the frozen VAE decoder at three matched resolutions: the deepest pixel tap is injected into the first upsampling stage, the intermediate tap into the second stage, and the shallowest tap into the third stage. The pixel extractor uses progressively wider channels across scales, and the latent-resolution pixel feature is projected to match the latent conditioning width before gated fusion. For image-space supervision, we apply the low-noise gating described in Sec. 3.3, so that pixel and perceptual losses are only applied when the predicted clean latent is sufficiently stable.

Degradation settings. In the clean paired stage, the model is trained with paired bicubic $\times 4$ LR–HR data rather than online real-world degradation. In the real-world stage, we use a second-order degradation pipeline following Real-ESRGAN. The pipeline includes stochastic blur, resizing, Gaussian or Poisson noise, JPEG compression, optional second degradation, final sinc filtering, and unsharp masking of the HR target. We use this stage to expose PGSR to realistic blur–noise–compression artifacts while preserving the same pixel-grounded architecture.

Inference. Unless otherwise specified, PGSR uses 20 denoising steps at inference time. We use 20 denoising steps and set the denoising strength to 1.0, i.e., sampling starts from the noisiest restoration state and follows the full restoration trajectory. The final latent prediction is decoded with the DFM-enabled VAE decoder. For large images, we use tiled evaluation with a default tile size of 512 and overlap of 64 to reduce memory usage while avoiding visible boundary artifacts.

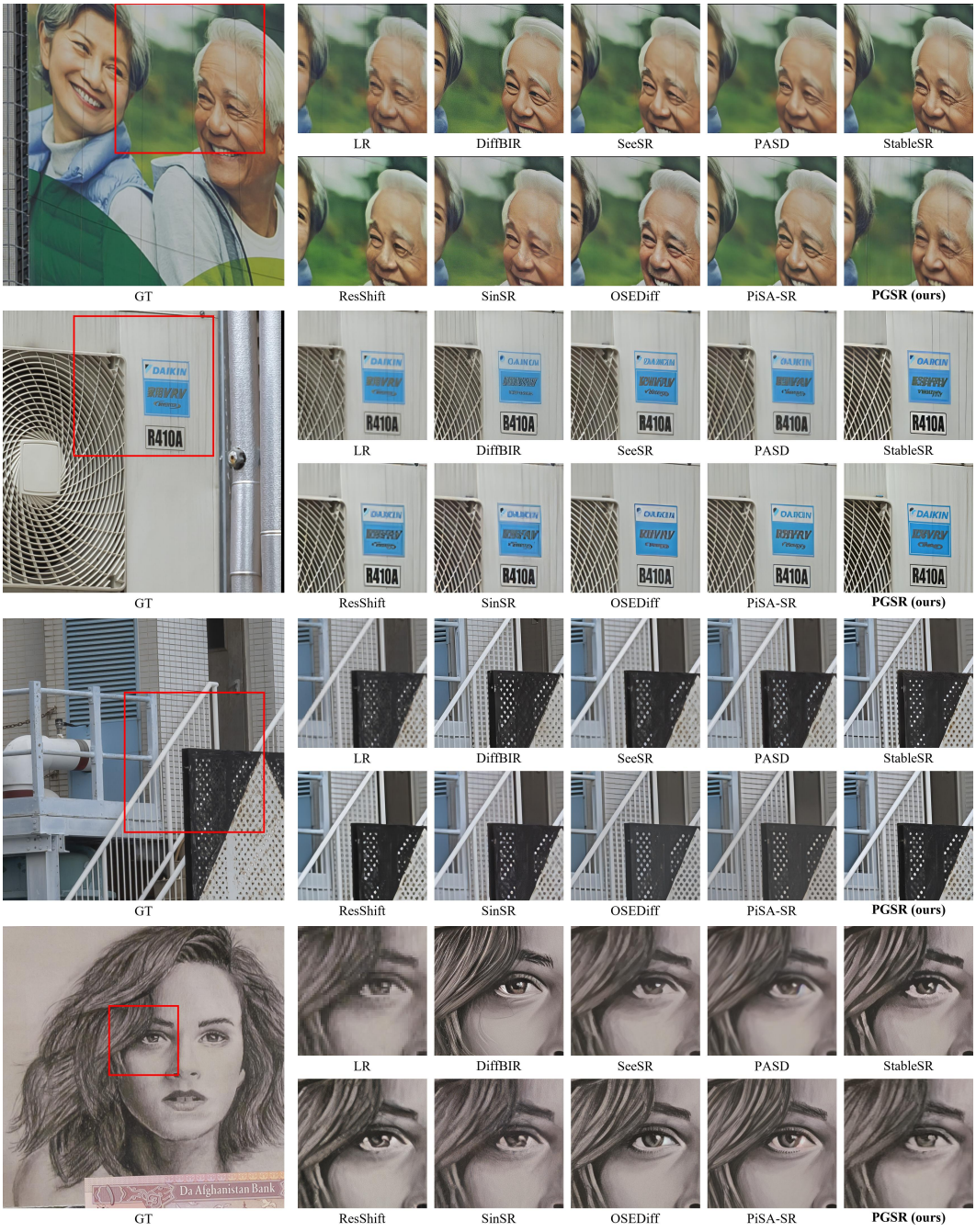


Figure 6: Additional qualitative comparison on real degradation.

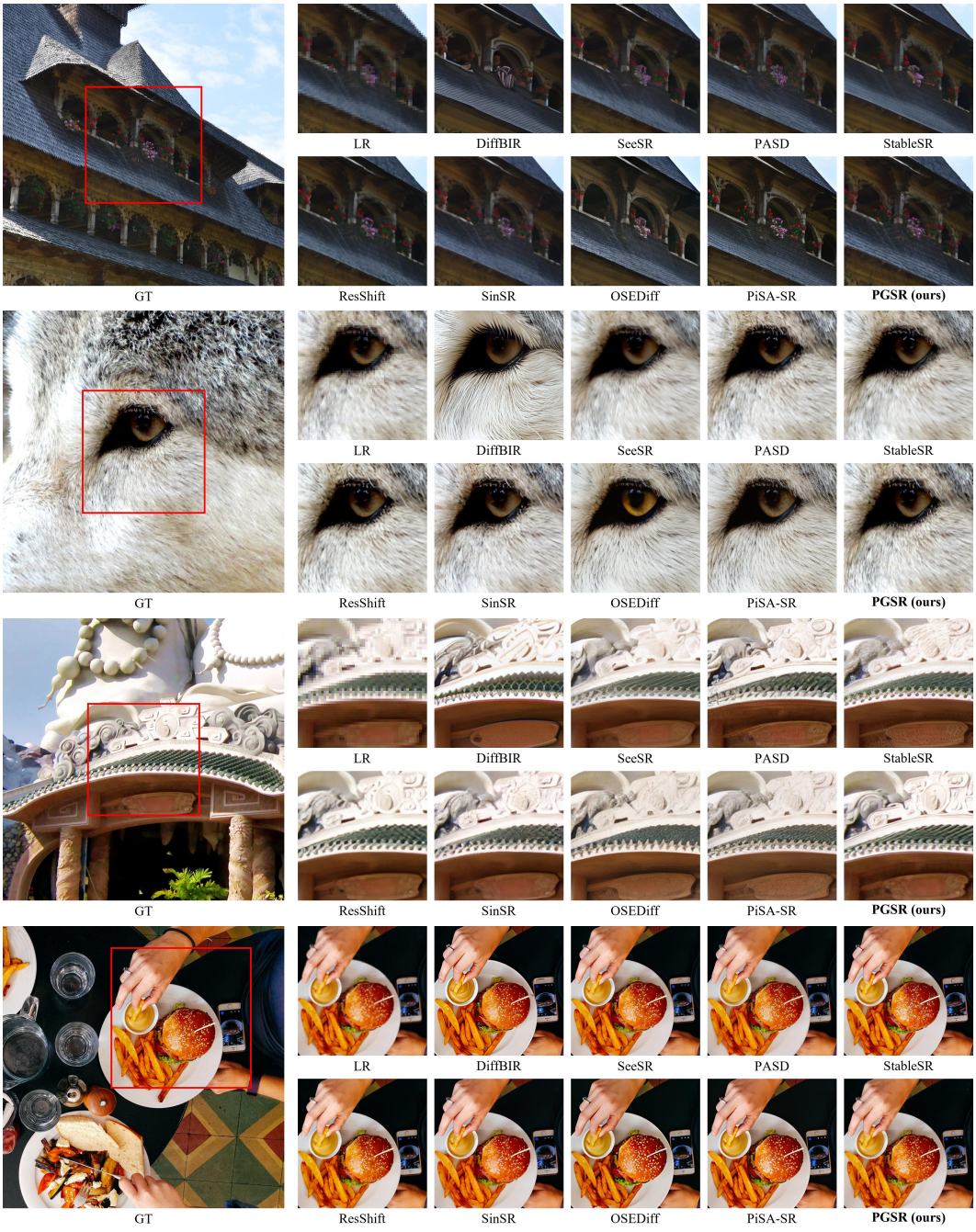


Figure 7: Additional qualitative comparison on synthetic degradation.

B Supplementary Experiments

B.1 Additional Visual Comparisons

We provide additional qualitative comparisons on RealSR, DRealSR, and DIV2K-Val in Figs. 6–7. These examples complement the main-text visualization by covering more diverse textures, structures, and real-world degradations.

B.2 Additional Ablation Tables

In this section, we present additional ablation studies on key hyperparameters and loss components.

Table 4 studies the most relevant training-time hyperparameters around the final Stage-2 recipe. We vary the perceptual loss weight λ_{perc}, the pixel-grounded reconstruction weight λ_{pg}, and the pixel-conditioning strength s_{pc}. Here s_{pc} scales the condition-side pixel evidence before it is fused into the LR latent condition; it is a conditioning strength and is distinct from the LR-anchored sampling strength used at inference time, which controls the denoising start point. Other optimizer and engineering settings are kept in the released configuration files.

Table 4: Sensitivity to the two image-space loss weights and the pixel-conditioning strength. Rows are grouped by the factor being swept; within each group, all other settings are fixed to the default Stage-2 recipe. We report PSNR and LPIPS to show the fidelity–perception effect of each hyperparameter.

Sweep	Variant	λ_{perc}	λ_{pg}	s_{pc}	DIV2K-Val		RealSR	
					PSNR↑	LPIPS↓	PSNR↑	LPIPS↓
Loss weights	No image-space loss	0	0	1.0	24.46	0.2301	23.25	0.2784
Loss weights	Perceptual only	0.05	0	1.0	24.68	0.2172	24.01	0.2635
Loss weights	Weak pixel loss	0.05	0.025	1.0	25.06	0.2113	24.53	0.2619
Loss weights	**Default**	**0.05**	**0.05**	**1.0**	**25.89**	**0.2104**	**24.75**	**0.2596**
Loss weights	Strong pixel loss	0.05	0.10	1.0	25.83	0.2160	24.32	0.2655
Conditioning	Weak pixel conditioning	0.05	0.05	0.8	26.01	0.2307	24.67	0.2706
Conditioning	Strong pixel conditioning	0.05	0.05	1.2	25.42	0.2244	24.38	0.2664

Table 5 reports the loss-component ablation used to separate supervision effects from the architectural ablation in the main text. Starting from the latent flow-matching objective, adding perceptual supervision improves LPIPS on both DIV2K-Val and RealSR, while the pixel-grounded reconstruction loss further improves distortion metrics. We also test an optional perceptual term on the DFM-decoded prediction to examine whether the decoder-side pixel-grounding branch benefits from perceptual supervision in addition to pixel-wise reconstruction:

$$\mathcal{L}_{\mathrm{ext}} = \mathcal{L}_{\mathrm{fm}} + \lambda_{\mathrm{perc}}\mathcal{L}_{\mathrm{perc}} + \lambda_{\mathrm{pg}}\mathcal{L}_{\mathrm{pg}} + \lambda_{\text{pg-perc}}\mathcal{L}_{\text{pg-perc}}, \quad \mathcal{L}_{\text{pg-perc}} = m_{\mathrm{pg}}\mathcal{L}_{\mathrm{LPIPS}}(\hat{x}_{hr}^{\mathrm{DFM}}, x_{hr}).$$

Here $\hat{x}_{hr}^{\mathrm{DFM}}$ is decoded through the DFM path and m_{pg} uses the same low-noise gate as $\mathcal{L}_{\mathrm{pg}}$. The default PGSR recipe sets $\lambda_{\text{pg-perc}} = 0$; the final row is included only as an exploratory ablation of perceptual supervision inside the DFM branch.

Table 5: Loss-component ablation of PGSR on DIV2K-Val and RealSR validation. Each row progressively adds one supervision term to the latent flow-matching baseline. The optional DFM perceptual term tests whether applying LPIPS directly on the DFM-decoded prediction further improves perceptual quality.

Variant	$\mathcal{L}_{\mathrm{fm}}$	$\mathcal{L}_{\mathrm{perc}}$	$\mathcal{L}_{\mathrm{pg}}$	$\mathcal{L}_{\text{pg-perc}}$	DIV2K-Val			RealSR		
					PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
FM control only	✓	–	–	–	24.67	0.7580	0.2431	22.15	0.7305	0.2789
+ perceptual	✓	✓	–	–	24.78	0.7651	0.2188	24.08	0.7401	0.2626
PGSR (default)	✓	✓	✓	–	**25.89**	**0.7707**	**0.2104**	**24.75**	0.7495	0.2596
+ DFM perceptual	✓	✓	✓	✓	25.70	0.7696	**0.2102**	24.36	0.7490	**0.2588**

B.3 Transfer Across Generative Backbones

We evaluate whether the benefit of pixel grounding extends beyond the FLUX backbone by implementing it on an SD2.1-based SR model. Table 6 compares each backbone's latent-only SR baseline with its pixel-grounded counterpart on DIV2K-Val. On SD2.1, pixel grounding improves PSNR by 2.40 dB and SSIM by 0.0605, while reducing LPIPS from 0.2591 to 0.2316. The corresponding FLUX improvements are 2.21 dB, 0.0115, and 0.0327, respectively. These within-backbone comparisons support the contribution of pixel grounding beyond the choice of pretrained generative backbone.

Table 6: Pixel grounding improves both distortion and perceptual metrics on DIV2K-Val with SD2.1 and FLUX. Bold denotes the better result within each backbone.

Backbone	Method	PSNR↑	SSIM↑	LPIPS↓
SD2.1	Latent-only SR baseline	21.33	0.7048	0.2591
SD2.1	Baseline + pixel grounding	**23.73**	**0.7653**	**0.2316**
FLUX.1-dev	Latent-only SR baseline	23.68	0.7592	0.2431
FLUX.1-dev	PGSR	**25.89**	**0.7707**	**0.2104**

The transfer changes both the generative backbone and its native VAE tokenizer. Both tokenizers use an $f8$ spatial downsampling ratio, so this experiment evaluates transfer across backbones and tokenizers rather than across compression ratios. The magnitude of the gain varies between the two systems; the results do not imply equal effectiveness for arbitrary architectures.

B.4 Training and Inference Cost

Table 7 compares PGSR with SinSR, OSEDiff, StableSR, and LucidFlux [Fei et al., 2025]. Inference timings use the same GPU, batch size 1, and a $448 \times 448 \rightarrow 1792 \times 1792$ SR task. We synchronize CUDA, exclude model loading, use one warm-up image, and average the next ten images. SinSR, OSEDiff, and StableSR use FP16; LucidFlux and PGSR use BF16, following the evaluated implementations. LucidFlux uses its official 20-step configuration and serves as a FLUX-based universal-restoration reference.

Table 7: Measured inference latency and training cost. Time/step is the amortized time per image divided by the sampling-step count. Memory is peak training memory per GPU. LucidFlux is a universal-restoration model; the other methods are evaluated as SR systems.

Method	Backbone	Steps	Time/step	Time/image	Train time/iter.	Peak train mem./GPU
SinSR	ResShift	1	2.14 s	2.14 s	0.67 s	5.65 GiB
OSEDiff	SD2.1	1	9.79 s	9.79 s	1.11 s	23.65 GiB
StableSR	SD2.1	20	4.58 s	91.66 s	1.27 s	14.02 GiB
LucidFlux	FLUX.1-dev	20	1.23 s	24.61 s	4.26 s	35.54 GiB
PGSR	FLUX.1-dev	20	3.09 s	61.81 s	3.59 s	35.73 GiB
PGSR + sparse attention	FLUX.1-dev	20	1.95 s	38.95 s	1.89 s	39.14 GiB

PGSR remains slower per image than the one-step methods. Sparse attention reduces its inference latency from 61.81 s to 38.95 s (37.0%) and training time per iteration from 3.59 s to 1.89 s, but increases peak training memory from 35.73 GiB to 39.14 GiB. Thus, the measured speed improvement does not imply a reduction in memory use. The training figures include the cost of decoding predictions for image-space supervision: although the VAE weights are frozen, backpropagation through the decoder still requires computation and activation memory. Per-iteration measurements describe the evaluated training configurations and do not establish equal total training budgets across methods.

Condition-side grounding and final decoder modulation are compatible in principle with fewer sampling steps. However, a distilled one-step PGSR model has not been evaluated, and its latency or reconstruction quality cannot be inferred from the amortized time/step alone.

B.5 Incremental Computational Cost of Pixel Grounding

We profile the latent-only FLUX SR pipeline and its pixel-grounded variants under a $448 \times 448 \rightarrow 1792 \times 1792$ protocol with 20 sampling steps and 16 tiles of size 512×512. One multiply–add counts as two FLOPs. Table 8 counts the LR pixel extractor and condition fusion once per tile, with their outputs reused across sampling steps.

Table 8: Inference FLOPs under a shared tiled-evaluation protocol. Relative changes use the latent-only FLUX SR pipeline as the baseline. These operation counts quantify incremental grounding cost rather than wall-clock acceleration.

Variant	TFLOPs/image	Relative to baseline
Latent-only FLUX SR baseline	7587.455	0%
Condition-side only	7587.559	$+0.0014\%$
Decoder-side only	7588.899	$+0.0190\%$
Full PGSR	7588.899	$+0.0190\%$
Full PGSR + sparse attention	7579.106	-0.1100%

Full pixel grounding adds 1.444 TFLOPs per image, or approximately 0.019% of the baseline cost. Decoder-side-only and full PGSR have identical rounded counts because they share the pixel extractor and the additional fusion cost is negligible at this scale. For comparison, an evaluator that recomputes the extractor and fusion at every denoising step requires 7590.862 TFLOPs/image ($+0.0449\%$). The small incremental cost under both accounting choices indicates that repeated FLUX sampling dominates the operation count. The sparse-attention FLOPs reduction is much smaller than its measured latency reduction in Table 7; FLOPs and runtime therefore describe different aspects of efficiency.

B.6 Decoder Feature Sources and Modulation

We study the Pixel Extractor + DFM design through a controlled decoder-only reconstruction test. All variants receive the same clean-HR-derived latent from the FLUX VAE, removing differences in the flow trajectory. The VAE encoder and decoder remain frozen. We compare the native decoder, a CFW-style fusion reference adapted from StableSR, DFM driven by intermediate VAE-encoder features, and DFM driven by PGSR's independent pixel extractor. CFW-style is a reference modulation operator, rather than a component proposed by PGSR or a reproduction of StableSR's complete SR pipeline.

Table 9: Decoder-only reconstruction comparison. Parameters count only trainable modules; the frozen VAE is excluded. The final row includes both the pixel extractor and DFM. Bold denotes the best reconstruction metric.

Variant	Auxiliary feature	Trainable params.	Rec. PSNR↑	LPIPS↓	DISTS↓
Frozen FLUX VAE	None	0.00 M	33.30	0.0190	0.0056
Frozen VAE + CFW-style	VAE encoder taps	105.82 M	35.66	0.0222	0.0116
Frozen VAE + DFM	VAE encoder taps	4.92 M	36.38	0.0157	0.0089
Frozen VAE + Pixel Extractor + DFM	Pixel-extractor taps	2.85 M	**38.09**	**0.0125**	**0.0051**

With identical VAE-encoder taps and decoder injection sites, replacing CFW-style fusion with DFM improves reconstruction PSNR from 35.66 dB to 36.38 dB and reduces LPIPS/DISTS from 0.0222/0.0116 to 0.0157/0.0089, using approximately $21.5\times$ fewer trainable parameters. The complete Pixel Extractor + DFM design further reaches 38.09 dB, 0.0125 LPIPS, and 0.0051 DISTS. This is the best reconstruction result among the tested variants and uses the fewest parameters among the trainable alternatives.

The 2.85 M count includes the pixel extractor (approximately 0.436 M) and DFM (2.412 M). The $32/64/256$-channel pixel taps require smaller adapters than the $128/256/512$-channel VAE taps (4.918 M). Thus, this comparison evaluates the combined feature-extraction and adapter design, not feature source at fixed width. The conclusions concern decoder reconstruction rather than end-to-end SR.

C Supplementary Analytical Discussion

This section provides additional analysis for the pixel-grounded components introduced in Sec. 3. The goal is not to establish global optimality, but to clarify the inductive biases behind the proposed design: gated residual condition fusion can attenuate unreliable pixel corrections while keeping the LR latent as an anchor, multi-scale pixel taps reduce the bottleneck imposed by a single flat pixel representation, and zero-initialized DFM preserves the frozen decoder before image-space supervision becomes reliable.

C.1 Gated Residual Condition Fusion

In Sec. 3, the condition-side fused latent is written as

$$\tilde{z}_{lr} = z_{lr} + \lambda\, \sigma(g)\, W_p(\bar{p}_{lr}),$$

where z_{lr} is the VAE-encoded LR condition, $\bar{p}_{lr}$ is the pre-VAE pixel feature projected to the latent resolution, W_p is a learned 1×1 projection, g is a learnable scalar gate, and λ controls the strength of the pixel evidence.

Proposition C.1. Assume that the ideal fused condition is $z_{lr} + \delta$, and the projected pixel correction is a noisy estimate $W_p(\bar{p}_{lr}) = \delta + \eta$ with $\mathbb{E}[\eta] = 0$ and $\mathbb{E}\|\eta\|_2^2 = \nu$. Among residual fusions of the form $z_{lr} + \lambda\gamma W_p(\bar{p}_{lr})$ with $\gamma \in [0, 1]$, the optimal gate is

$$\gamma^\star = \mathrm{clip}_{[0,1]}\left(\frac{\|\delta\|_2^2}{\lambda(\|\delta\|_2^2 + \nu)}\right),$$

and is no worse in expected squared error than any fixed additive choice when that fixed choice lies in the same interval.

Proof. For a given γ, the expected error of the fused condition is

$$\mathcal{E}(\gamma) = \mathbb{E}\,\|z_{lr} + \lambda\gamma(\delta + \eta) - (z_{lr} + \delta)\|_2^2 = (1 - \lambda\gamma)^2\|\delta\|_2^2 + \lambda^2\gamma^2\nu.$$

Minimizing this convex quadratic and projecting the minimizer onto $[0, 1]$ gives $\gamma^\star$. Since the gated family contains all fixed residual scales in $[0, 1]$, its best attainable error cannot be larger than that of direct addition with any such fixed scale. □

Discussion. This proposition should be read as a bias argument rather than a full optimization guarantee. The LR latent remains a stable residual anchor for the flow-matching trajectory, while the gate gives the model a simple mechanism to attenuate noisy pixel evidence under severe degradation. Compared with fixed additive fusion, gated residual fusion keeps the benefits of immediate pixel participation but avoids forcing every pixel correction to enter the latent condition with the same strength.

C.2 Approximation Advantage of Multi-Scale Pixel Taps

PGSR uses a three-stage pixel extractor that produces matched taps $\mathcal{S}_{lr} = \{s_1, s_2, s_3\}$ at $H/2$, $H/4$, and $H/8$ resolutions. This design is motivated by the decoder-side use case. The frozen latent decoder reconstructs an HR image through multiple upsampling blocks, and each block operates at a different spatial scale. A flat extractor that outputs only one latent-resolution feature must predict corrections for all decoder stages from the same compressed representation.

Let r_ℓ denote the ideal pixel correction for decoder stage ℓ. A flat extractor first maps the upsampled LR image into a single low-resolution feature q and then predicts each stage correction from q:

$$q = B_3 x_{lr}^{\uparrow}, \qquad \hat{r}_\ell^{\mathrm{flat}} = U_\ell A_\ell q,$$

where B_3 is the low-resolution pixel encoder, A_ℓ is a stage-specific projection, and U_ℓ upsamples the feature when required. The best achievable squared approximation error is

$$\mathcal{E}_{\mathrm{flat}} = \min_{\{A_\ell\}} \sum_\ell \mathbb{E}\left[\left\| r_\ell - U_\ell A_\ell B_3 x_{lr}^{\uparrow}\right\|_2^2\right].$$

By contrast, a multi-scale extractor predicts each correction from a matched feature tap:

$$s_\ell = B_\ell x_{lr}^{\uparrow}, \qquad \hat{r}_\ell^{\text{multi}} = A_\ell s_\ell,$$

with approximation error

$$\mathcal{E}_{\text{multi}} = \min_{\{A_\ell, B_\ell\}} \sum_\ell \mathbb{E}\left[\left\| r_\ell - A_\ell B_\ell x_{lr}^{\uparrow} \right\|_2^2\right].$$

Proposition C.2. The multi-scale formulation is at least as expressive as the flat formulation:

$$\mathcal{E}_{\text{multi}} \leq \mathcal{E}_{\text{flat}}.$$

Proof. The multi-scale family can emulate any flat solution by choosing each B_ℓ to factor through the same low-resolution representation $B_3 x_{lr}^{\uparrow}$ and absorbing the upsampling/projection into A_ℓ. Therefore, the hypothesis class optimized by $\mathcal{E}_{\text{flat}}$ is a subset of the hypothesis class optimized by $\mathcal{E}_{\text{multi}}$. Minimizing over a superset cannot yield a larger minimum error, which gives $\mathcal{E}_{\text{multi}} \leq \mathcal{E}_{\text{flat}}$. □

Discussion. This result should be interpreted as an approximation-capacity argument rather than a guarantee of better test performance. It supports the architectural choice of feeding DFM with scale-matched taps: high-frequency LR-observed cues may be attenuated by a latent-resolution bottleneck, and later projections cannot recover information that has already been discarded. Providing s_1, s_2, and s_3 gives the decoder access to pixel evidence at the same scales where details are rendered.

C.3 Identity Property of Decoder-Side DFM

For decoder stage ℓ, the DFM adapter applies

$$\tilde{f}_d^\ell = f_d^\ell + Z_\ell(\text{SiLU}(A_\ell(s_\ell))),$$

where f_d^ℓ is the frozen decoder activation, s_ℓ is the matched pixel feature tap, A_ℓ aligns channel width, and Z_ℓ is a zero-initialized output projection.

Proposition C.3. If Z_ℓ is zero-initialized, then DFM preserves the frozen decoder at initialization:

$$\tilde{f}_d^\ell = f_d^\ell \quad \text{for every decoder stage } \ell.$$

Proof. Since Z_ℓ is zero-initialized,

$$Z_\ell(\text{SiLU}(A_\ell(s_\ell))) = 0$$

at initialization for any input feature s_ℓ. Therefore,

$$\tilde{f}_d^\ell = f_d^\ell + 0 = f_d^\ell.$$

□

Discussion. The DFM-enhanced decoder starts exactly as the frozen VAE decoder and gradually learns pixel-grounded rendering only when image-space supervision provides a useful signal. This is important because the decoder is a pretrained component: without an identity-preserving initialization, injected pixel modulation could perturb decoder activations before the adapters learn meaningful spatial corrections.