# FLICKERDIFF: UNCERTAINTY-CONTROLLED ONE-STEP DIFFUSION FOR FLICKER REMOVAL

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Spatially non-uniform degradation makes it difficult for image restoration models to balance fidelity and generative recovery. Single-image flicker removal is a representative example of this problem. Flicker bands can severely attenuate text, edges, and object structures in some regions while leaving nearby content largely intact. Existing diffusion-based restoration methods often apply a uniform noise level or restoration strength across the image. They may therefore under-recover severely degraded regions or unnecessarily modify reliable content. We propose FlickerDiff, a one-step pixel-space diffusion model that adjusts this balance according to local reliability. An uncertainty estimator uses a frozen Restormer and its input–output discrepancy to construct a reliability map that determines the spatially varying transport variance in the flow-matching endpoint distribution. Reliable regions follow narrow paths anchored to the observation, while severely degraded regions receive a larger stochastic transport budget and can rely more strongly on the pretrained generative prior. The same uncertainty map is injected into the generator through token-aligned FiLM modulation. We adapt a pretrained 8B-parameter pixel-space Diffusion Transformer in its velocity-prediction space to build the one-step generator. Adversarial detail supervision and a lightweight post-fusion refiner further improve perceptual quality and output fidelity. Experiments show that FlickerDiff achieves the strongest reconstruction fidelity and perceptual quality among the compared methods. At inference, the full pipeline consists of uncertainty estimation, a single Pixel-DiT forward pass, and lightweight post-fusion refinement, without iterative diffusion sampling.




Figure 1: Illustrative example of spatially non-uniform flicker removal. The uncertainty map highlights the unreliable band and illustrates how FlickerDiff allocates stronger restoration to the degraded region while preserving nearby reliable content. OSEDiff and Restormer outputs are shown for context.



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

# 1 INTRODUCTION

Flicker and banding commonly appear when cameras capture scenes under temporally varying illumination, such as LED lighting, stage lights, billboards, or display screens, with mismatched exposure, rolling shutter, and refresh frequencies. Recent screen-capture restoration studies show that display-induced flicker-banding can strongly degrade readability and may coexist with moiré patterns in real captures (Zhu et al., 2025; 2026). Our setting is related but distinct: we target single images corrupted by scene-level illumination flicker and broad banding, where artifacts affect natural scene content rather than only a photographed screen area. Existing deflickering methods often rely on multiple frames, temporal alignment, or video statistics. In contrast, single-image flicker removal must infer a clean image from one corrupted observation, making it relevant to still photos, extracted video frames, archival footage, and non-repeatable capture scenarios.

Severe flicker creates a local completion–fidelity trade-off: reliable regions should remain close to the input, whereas heavily attenuated text, edges, and object parts require a generative prior. Uniform restoration strength can either leave severe bands incomplete or unnecessarily alter reliable content. We use single-image flicker removal to study spatial control of this trade-off.

One-step diffusion restoration provides a promising route to this problem. OSEDiff and related score-distillation methods reduce the sampling cost of pretrained diffusion priors for restoration (Wu et al., 2024). FideDiff argues that high-fidelity one-step restoration should preserve a restoration-specific trajectory rather than collapse diffusion into a crude fixed-timestep regression model (Liu et al., 2026). ODTSR shows that a large Diffusion Transformer (DiT) can be adapted for one-step real-world image super-resolution by explicitly reasoning about fidelity and controllability (Fang et al., 2025). These works suggest that the diffusion endpoint is not merely an implementation detail: its noise level affects how strongly the pretrained prior participates in restoration. For flicker removal, this control should be spatial rather than only global. The desired generation strength should follow local degradation uncertainty.

We therefore formulate severe single-image flicker removal around a simple question: how should an uncertainty map control the amount of diffusion prior used at each spatial location? Given a low-quality image and an uncertainty map produced by an uncertainty estimator built on a frozen Restormer, FlickerDiff adapts a pretrained pixel image-editing DiT into a one-step restorer. Instead of only concatenating the map as an input condition or using it as a loss mask, we use the map to modulate the endpoint noise distribution. This directly matches the task asymmetry: low-uncertainty regions stay close to the input for fidelity, while high-uncertainty regions receive stronger stochastic perturbation for local completion.

Our contributions are:

- We formulate severe single-image flicker removal as spatially non-uniform completion–fidelity control under uneven observability, rather than as uniform artifact suppression.
- We introduce uncertainty-guided heteroscedastic endpoint modulation for one-step flow-matching/Pixel-DiT restoration. The uncertainty map defines the endpoint distribution itself, which differs from loss weighting, simple concatenated conditioning, and global noise or timestep control.
- We provide evidence that uncertainty affects both output behavior and hidden DiT computation: global and region metrics, fixed/uniform/random/inverted map ablations, high-U/low-U evaluation, hidden-token response analysis, PCA, and causal hidden-token intervention.

# 2 RELATED WORK

## 2.1 FLICKER AND BANDING REMOVAL

Flicker removal has been studied in video restoration and burst restoration settings, where multiple frames help distinguish true scene structure from periodic illumination artifacts. BurstDeflicker introduces a real captured benchmark for flicker removal in dynamic scenes (Qu et al., 2025). Its follow-up Flickerformer, titled *It Takes Two*, explicitly exploits periodicity and directionality across

burst frames for burst flicker removal (Qu et al., 2026). These burst methods solve a different problem from ours: they can use neighboring corrupted frames and cross-frame redundancy, while we restore each low-quality image independently. We use the BurstFlicker-S single-image protocol derived from this benchmark, where severe bands must be removed without temporal or burst cues. This setting is particularly challenging because severe bands can locally erase text, edges, and object structures, making the task closer to uncertainty-aware completion than uniform artifact removal. DeflickerCycleGAN studies a related single-image flicker removal setting with a cycle-consistent adversarial framework (Lin et al., 2023).

More recently, RIFLE formulates screen-image flicker-banding removal as a dedicated restoration problem and introduces a latent diffusion enhancement framework with a flicker-banding prior estimator and masked loss; it uses a synthetic flicker-banding pipeline for training and a paired real-world screen-capture benchmark for evaluation (Zhu et al., 2025). CLEAR studies a broader compound screen-capture setting where flicker-banding and moiré artifacts coexist, constructing the MIRAGE screen-capture dataset and introducing an ISP-based flicker simulation pipeline to stabilize and expand training degradations (Zhu et al., 2026). These works highlight the practical importance of structured banding, but their primary degradation source is screen recapture: aliasing between an emissive display and a camera sensor, often coupled with screen moiré. In contrast, BurstFlicker-S contains environment-light-induced global or wide-area flicker bands over natural scenes. Our method also uses a different mechanism: an estimated uncertainty map spatially modulates the endpoint noise distribution of a one-step pixel-space DiT, rather than estimating display band parameters or solving joint moiré–banding removal.

### 2.2 Diffusion Models for Image Restoration

Diffusion models have become strong priors for real-world image restoration and super-resolution. Latent Diffusion Models provide the foundation for Stable-Diffusion-based restoration pipelines (Rombach et al., 2022). Flow Matching and Rectified Flow formulate generation as learning a velocity field along a probability path, enabling efficient deterministic integration and one-step approximations (Lipman et al., 2023; Liu et al., 2023). ResShift performs efficient residual-shifting diffusion in pixel space (Yue et al., 2023); OSEDiff adapts pretrained Stable Diffusion priors for perceptual restoration (Wu et al., 2024); PiSA-SR, SinSR, and StrSR further explore one-step or structure-aware super-resolution pipelines (Sun et al., 2025; Wang et al., 2024; 2026). Closely related to our setting, FideDiff studies efficient diffusion for high-fidelity motion deblurring and emphasizes that one-step restoration should preserve the restoration objective rather than only optimize perceptual realism (Liu et al., 2026). These methods provide strong baselines and useful design principles, but most are designed for super-resolution, deblurring, or generic degradation rather than severe single-image flicker and banding.

### 2.3 Diffusion Transformers for One-Step Restoration

Diffusion Transformers (DiTs) have become a scalable alternative to convolutional diffusion backbones (Peebles & Xie, 2023). DiT-based image foundation models such as Qwen-Image show strong text rendering, editing, and structure generation ability (Wu et al., 2025). ODTSR adapts Qwen-Image into a one-step Real-ISR model using a noise-hybrid visual stream and fidelity-aware adversarial training (Fang et al., 2025). We follow the broader insight that endpoint noise controls the trade-off between fidelity and generation, but use it differently: our control is spatial and degradation-driven, rather than a global fidelity weight for prompt controllability.

### 2.4 Uncertainty-Guided Restoration

Uncertainty has been used in restoration as a spatial reliability signal in several ways. Prior work has used uncertainty or degradation estimates for loss weighting, quality/reliability estimation, perturbation control, and sampling guidance in super-resolution and restoration (Ning et al., 2021; Zhang et al., 2025; He et al., 2025; Yin et al., 2026). These ideas are closely related to our motivation: unreliable regions should not be treated the same as reliable regions. Our mechanism is different in where the uncertainty enters the generative process. The estimated map defines the endpoint distribution itself for a fixed-time one-step flow-matching path; it is not only a supervision weight, not only an auxiliary condition, and not only a guidance signal inside an iterative sampler. We also

test whether the effect is spatially aligned rather than merely caused by increased global noise energy, using uniform-energy, random, and inverted map controls. Finally, we provide hidden-token localization and causal token-swap evidence to probe whether the uncertainty-controlled endpoint changes the DiT computation in the intended regions.

# 3 METHODOLOGY

## 3.1 PROBLEM SETUP AND OVERVIEW

Given a single flicker-corrupted image $I_{\mathrm{LQ}} \in \mathbb{R}^{3\times H\times W}$, the goal is to recover a clean image $I_{\mathrm{GT}}$ while preserving scene geometry, color, text, and reliable regions. Unlike video deflickering, no neighboring frames are available. The model must infer from one observation which regions should remain anchored to the input and which regions require stronger restoration.

We use an estimated uncertainty map $B \in [0,1]^{1\times H\times W}$ as this spatial reliability signal. Larger $B_{ij}$ means that location $(i,j)$ is less reliable and more likely to be affected by severe flicker or banding. The map is available at test time and is not computed from the ground-truth residual. It does not reveal clean content; it only indicates where the observation should be trusted less.

This setup creates the central design requirement of FlickerDiff: the one-step diffusion restorer should not use the same amount of generative freedom everywhere. Low-uncertainty pixels should remain close to $I_{\mathrm{LQ}}$, while high-uncertainty pixels should receive a stronger diffusion prior for local completion.

FlickerDiff implements this requirement with three components. First, an uncertainty estimator derives a spatial reliability map from the discrepancy between a frozen Restormer output and the input. Second, a pretrained pixel-space Diffusion Transformer is adapted into a one-step restorer. Third, the uncertainty map controls the endpoint noise distribution used by the one-step flow-matching objective and is also exposed to the DiT through lightweight token-aligned FiLM conditioning. We additionally study adversarial detail supervision and direct post-fusion as refinement stages. Unless stated otherwise, the deployed model uses uncertainty-guided endpoint modulation, FiLM4 conditioning, adversarial detail supervision during generator training, and a lightweight direct region-loss post-fusion refiner. The discriminator is discarded after training, and inference consists of one Pixel-DiT pass followed by the refiner.

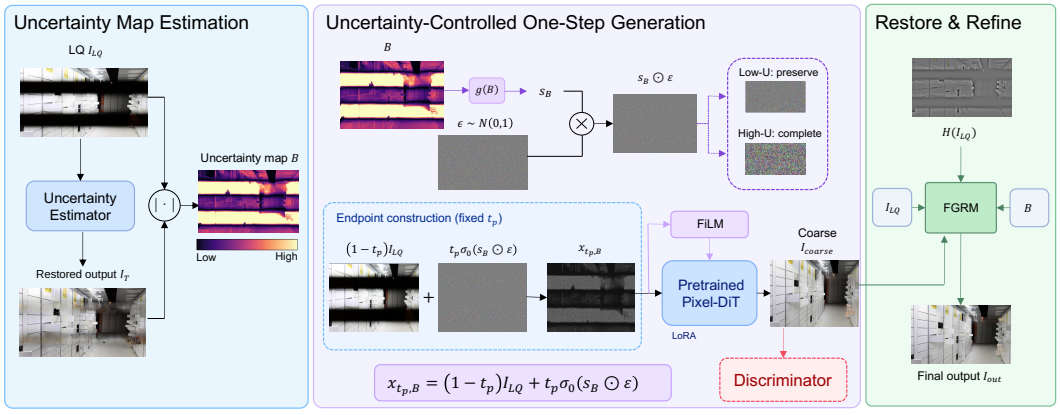


Figure 2: Overview of FlickerDiff. An uncertainty estimator built on a frozen Restormer derives the reliability map from the flicker-corrupted input. The map controls the spatial endpoint noise distribution and is provided to the one-step Pixel-DiT through token-aligned FiLM modulation. Adversarial supervision is used during generator training, while the lightweight fusion refiner restores output fidelity at deployment.

### 3.2 ONE-STEP PIXEL-DIT FOUNDATION RESTORER

The uncertainty map requires a generator whose behavior can be controlled through the diffusion endpoint. We therefore adapt the pretrained HiDream-O1 pixel-space image-editing DiT into a one-step restorer (Cai et al., 2026). Unlike latent restoration pipelines, the generator operates directly in pixel space, avoiding VAE compression artifacts in the restored RGB output and preserving fine text and edges. The low-quality image is provided as the editing condition, and the trainable parameters are restricted to LoRA adapters unless otherwise specified.

Under the flow-matching parameterization, a straight probability path has the form $x_t = (1-t)x_0 + tx_1$ and its sample-wise velocity is $x_1 - x_0$ (Lipman et al., 2023; Liu et al., 2023). We retain this output space instead of replacing the model with a direct regression head, because the velocity field provides a natural interface for controlling how much stochastic transport is assigned to each region. A spatially uniform endpoint baseline constructs:

$$x_{t_p} = (1 - t_p)I_{\mathrm{LQ}} + t_p\sigma_0\epsilon, \quad \epsilon \sim \mathcal{N}(0, I). \tag{1}$$

To make the one-step target explicit, consider the local conditional bridge between the clean target and this reverse-flow starting state:

$$x_s = \left(1 - \frac{s}{t_p}\right) I_{\mathrm{GT}} + \frac{s}{t_p} x_{t_p}, \quad 0 \le s \le t_p. \tag{2}$$

Its sample-wise velocity is constant, $v^\star = (x_{t_p} - I_{\mathrm{GT}})/t_p$. The model predicts this velocity at $t_p$ and reconstructs the restored image with one reverse Euler update:

$$\hat{x}_0 = x_{t_p} - t_p v_\theta(x_{t_p}, I_{\mathrm{LQ}}, t_p, c), \tag{3}$$

where $c$ is a task prompt such as removing flicker bands while preserving content. Importantly, velocity regression and endpoint reconstruction are equivalent at the fixed time:

$$\|v_\theta - v^\star\|_2^2 = \frac{1}{t_p^2}\,\|\hat{x}_0 - I_{\mathrm{GT}}\|_2^2\,. \tag{4}$$

This formulation preserves the pretrained flow-matching output space while reducing restoration to a single endpoint prediction. However, a spatially uniform endpoint assigns the same generation strength to all regions, which can over-perturb clean areas and under-activate the prior inside severe bands.

### 3.3 UNCERTAINTY-GUIDED ENDPOINT CONTROL

**Map construction.** We construct an uncertainty estimator using a frozen Restormer (Zamir et al., 2022). Let $I_T = T_\eta(I_{\mathrm{LQ}})$ be its restored image. The normalized discrepancy between $I_T$ and the input serves as a spatial reliability proxy. Since $T_\eta$ is trained for fidelity-oriented restoration, large local changes between $I_T$ and $I_{\mathrm{LQ}}$ indicate regions where the observation is likely corrupted and where even conservative restoration needs to modify the input. The raw RGB discrepancy is:

$$U_{\mathrm{rgb}}(p) = \frac{1}{3}\sum_{c=1}^{3} \left|I_T^c(p) - I_{\mathrm{LQ}}^c(p)\right|, \tag{5}$$

$$B(p) = G_k\left(\mathrm{clip}\left(\frac{U_{\mathrm{rgb}}(p)}{\phi_{\max}}, 0, 1\right)\right), \tag{6}$$

where $\phi_{\max}$ is a normalization bound and $G_k$ is an optional Gaussian smoothing operator. All main experiments use this raw RGB Restormer–input difference with $\phi_{\max} = 0.4$. This construction deliberately avoids $I_{\mathrm{GT}}$: the map does not reveal clean content, but only predicts where the LQ observation should be trusted less.

On the BurstFlicker-S dataset, the image-level estimated uncertainty score is strongly correlated with the image-level ground-truth residual score, with Pearson $r = 0.873$ and Spearman $\rho = 0.912$ when both scores are computed as the mean of the top 10% pixels; see Appendix A.4. This supports using the estimated map as a degradation-reliability proxy while keeping the inference pipeline independent of ground-truth information.

**Endpoint modulation.** Let $g(B)$ denote the map-to-noise transform. The selected main variant directly clamps the raw uncertainty map:

$$\mathbf{s}_B = g(B) = \mathrm{clip}(B, s_{\min}, s_{\max}), \tag{7}$$

with $s_{\min} = 0.2$ and $s_{\max} = 1.0$ in the main configuration. The endpoint is then constructed as:

$$x_{t_p,B} = (1 - t_p)I_{\mathrm{LQ}} + t_p \sigma_0 \mathbf{s}_B \odot \epsilon. \tag{8}$$

The corresponding velocity target is:

$$v^\star = \frac{x_{t_p,B} - I_{\mathrm{GT}}}{t_p}. \tag{9}$$

The flow interpretation also exposes how uncertainty allocates the stochastic part of the transport. At pixel $p$, the target velocity is

$$v^\star(p) = \frac{(1 - t_p)I_{\mathrm{LQ}}(p) - I_{\mathrm{GT}}(p)}{t_p} + \sigma_0 s_p \epsilon(p), \tag{10}$$

and therefore, for $C$ image channels,

$$\mathbb{E}_\epsilon\left[\|v^\star(p)\|_2^2\right] = \left\| \frac{(1 - t_p)I_{\mathrm{LQ}}(p) - I_{\mathrm{GT}}(p)}{t_p} \right\|_2^2 + C\sigma_0^2 s_p^2. \tag{11}$$

The uncertainty map thus allocates a spatial stochastic transport budget: reliable pixels follow a narrower, more observation-anchored path, while uncertain pixels receive a larger stochastic velocity component and more freedom for local completion. This is a fixed-time heteroscedastic flow endpoint, not a spatially varying timestep. The training objective becomes:

$$\mathcal{L}_{\mathrm{endpoint}} = \mathbb{E}_{I_{\mathrm{LQ}}, I_{\mathrm{GT}}, B, \epsilon}\left[ \left\| v_\theta(x_{t_p,B}, I_{\mathrm{LQ}}, t_p, c) - \frac{x_{t_p,B} - I_{\mathrm{GT}}}{t_p} \right\|_2^2 \right]. \tag{12}$$

At inference time, the same endpoint construction is used:

$$\hat{x}_0 = x_{t_p,B} - t_p v_\theta(x_{t_p,B}, I_{\mathrm{LQ}}, t_p, c). \tag{13}$$

Training and inference therefore share the same uncertainty-conditioned endpoint distribution. This is different from training with a weighted loss and then sampling from a fixed-noise endpoint.

This differs from simply concatenating $B$ as an input condition. The uncertainty map changes the training distribution seen by the denoiser: low-uncertainty regions are exposed to weaker stochastic perturbation and remain anchored to $I_{\mathrm{LQ}}$, while high-uncertainty regions are trained under stronger perturbation and can rely more on the pretrained image prior. In the ablation study, fixed, random smooth, inverted, and energy-matched uniform maps separate the effect of global noise magnitude from the effect of spatial alignment with real degradation uncertainty.

**Token-aligned conditioning.** Endpoint noise modulation is the primary use of uncertainty, but the denoiser can also benefit from seeing the same spatial reliability signal explicitly. Given the raw uncertainty map, we concatenate normalized image coordinates and embed the resulting map with a stride-$P$ convolution:

$$m = \mathrm{Conv}_{P \times P, P}([B, X, Y]) \in \mathbb{R}^{N \times d_m}, \tag{14}$$

where $N$ matches the number of image tokens. At a small set of selected transformer blocks, the map tokens modulate the image tokens through a zero-initialized FiLM-style residual:

$$[\gamma_\ell, \beta_\ell] = W_\ell \mathrm{LN}(m), \tag{15}$$

$$h_\ell \leftarrow h_\ell + r_\ell \odot \left(\gamma_\ell \odot \mathrm{LN}(h_\ell) + \beta_\ell\right), \tag{16}$$

where $h_\ell$ denotes the image-token hidden states and $r_\ell$ is a learnable residual gate. The scale–shift projection is initialized to zero, so the pretrained DiT initially behaves like the endpoint-only model and learns to use the map only when it improves restoration. We use four such FiLM layers in the selected coarse and adversarially trained generators, and isolate their effect in the ablation study. This module does not replace the uncertainty-guided endpoint distribution; it exposes the same reliability map as a lightweight local condition.

### 3.4 Auxiliary Detail Supervision and Refinement

To reduce over-smoothing, we train the coarse generator with endpoint reconstruction, $L_1$, LPIPS, and optional adversarial losses:

$$\mathcal{L}_G = \mathcal{L}_{\text{endpoint}} + 0.5\|\hat{x}_0 - I_{\text{GT}}\|_1 + \mathcal{L}_{\text{LPIPS}} + \lambda_{\text{adv}}\mathcal{L}^G_{\text{adv}}. \tag{17}$$

We use $\lambda_{\text{adv}} = 0$ for the reconstruction-trained coarse model and $0.01$ for adversarial polishing (Fang et al., 2025). A training-only discriminator combines frozen Wan VAE features, a trainable Wan DiT, and a PatchGAN head with relativistic supervision (Team Wan et al., 2025; Isola et al., 2017; Jolicoeur-Martineau, 2019).

At deployment, a lightweight NAFNet-style refiner (Chen et al., 2022) receives the coarse output, input, high-frequency input, and uncertainty map. Its RGB, perceptual, and gradient objectives are supplemented with uncertainty-aware terms that preserve coarse completions in High-U regions and input fidelity in Low-U regions. Appendix A.2 gives the architecture interface, losses, and weights. We evaluate endpoint control separately from these auxiliary stages.

## 4 Experiments

We evaluate FlickerDiff from three complementary angles: overall restoration quality, uncertainty-specific behavior, and the contribution of the endpoint mechanism. We first compare against representative restoration and diffusion baselines on the BurstFlicker-S dataset. We then report stratified and masked-region evaluations to test whether gains hold in high-uncertainty regions, followed by controlled ablations and representation analyses.

### 4.1 Experimental Settings

**Implementation.** We adapt the pixel-space HiDream-O1 generator with rank-32 LoRA for one-step restoration. Training uses $1024 \times 512$ paired crops, $t_p = 0.43$, $\sigma_0 = 8.0$, raw uncertainty clamp $[0.2, 1.0]$, and mixed precision. FiLM4 embeds uncertainty and coordinates into token-aligned conditions; a lightweight NAFNet-style refiner is trained after generator inference. Detailed settings are in Appendix A.1.

**Datasets.** We use the BurstFlicker-S single-image protocol derived from the real captured burst flicker benchmark (Qu et al., 2025); neighboring burst frames are withheld. The training split contains 350 scenes and 3,525 paired images, and all results use the official paired targets and evaluation split. Uncertainty maps are precomputed with the uncertainty estimator using the discrepancy in Sec. 3.3, without accessing ground truth.

**Metrics.** At $1024 \times 512$, we report PSNR, SSIM, LPIPS (Zhang et al., 2018), DISTS (Ding et al., 2022), MUSIQ (Ke et al., 2021), and MANIQA (Yang et al., 2022), all computed with PyIQA after paired RGB alignment.

**Compared methods.** We compare with representative single-image restoration and diffusion baselines, including DeflickerCycleGAN, Restormer, PiSA-SR, OSEDiff, ResShift, SinSR, StrSR-ZImage, and multi-step HiDream. Each method follows its original recipe or a stable BurstFlicker-S adaptation; outputs are evaluated at the same resolution and with the same PyIQA protocol. RIFLE and CLEAR are discussed as related screen-capture works, while the unadapted HiDream prior is reported in Appendix A.

### 4.2 Main Results

Table 1: Main comparison on the BurstFlicker-S dataset. We report the deployed output of FlickerDiff against representative single-image restoration and diffusion baselines.

| Method | PSNR↑ | SSIM↑ | LPIPS↓ | DISTS↓ | MUSIQ↑ | MANIQA↑ |
|---|---|---|---|---|---|---|
| DeflickerCycleGAN | 24.146 | 0.863 | 0.177 | 0.115 | 62.622 | 0.415 |
| Restormer | 25.190 | 0.888 | 0.130 | 0.097 | 63.442 | 0.416 |
| PiSA-SR | 24.698 | 0.829 | 0.143 | 0.113 | 64.011 | 0.415 |
| OSEDiff | 25.733 | 0.876 | 0.104 | <u>0.072</u> | 66.494 | <u>0.435</u> |
| ResShift | 26.819 | <u>0.904</u> | 0.118 | 0.090 | <u>66.846</u> | 0.417 |
| SinSR | 26.505 | 0.881 | 0.160 | 0.141 | 62.490 | 0.400 |
| StrSR-ZImage | 24.782 | 0.887 | 0.121 | 0.084 | 64.457 | 0.422 |
| HiDream multi-step | 26.616 | 0.870 | 0.141 | 0.121 | 62.449 | 0.366 |
| Ours | **28.998** | **0.914** | **0.067** | **0.062** | **67.776** | **0.437** |

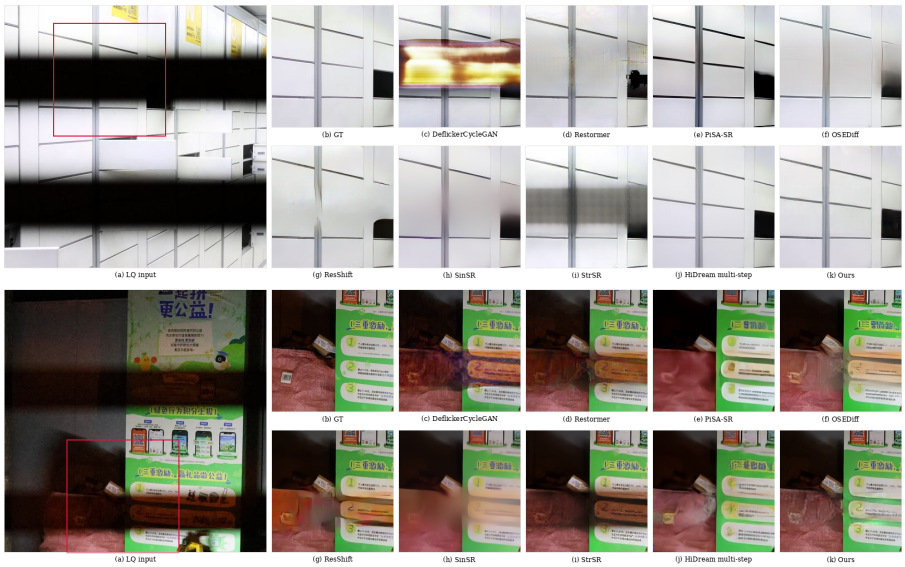


Figure 3: Qualitative comparison on severe cases from BurstFlicker-S. Each example shows a $512 \times 512$ input view on the left and matching square crops from the ground truth and restoration results on the right. Red boxes mark the crop regions.

The deployed FlickerDiff system achieves the strongest full-reference and no-reference scores on BurstFlicker-S. Figure 3 shows that conservative baselines leave severe regions incomplete, while generic diffusion baselines remain less reliable for paired fidelity; detailed component analyses are in Appendix A.

### 4.3 Ablation Study

**Spatial alignment.** We first isolate endpoint control using identical backbones, training budgets, and reconstruction losses, without LPIPS, adversarial supervision, or post-fusion. Table 2 compares fixed noise, energy-matched uniform noise, and spatial controls. Uniform energy matches the real map's global RMS noise; random and inverted maps disrupt its alignment with degradation.

Table 2: Controlled endpoint ablation on BurstFlicker-S under pure reconstruction training. High-U and Low-U are the top 20% and bottom 50% pixels of the real uncertainty map for every variant.

| Endpoint | PSNR↑ | LPIPS↓ | High-U PSNR↑ | Low-U PSNR↑ |
|---|---|---|---|---|
| Fixed noise | 26.639 | 0.207 | 25.557 | 28.168 |
| Uniform energy | 27.191 | 0.192 | 25.825 | 28.950 |
| Random smooth | 26.703 | 0.209 | 25.539 | 28.351 |
| Inverted map | 27.074 | 0.192 | 25.924 | 28.695 |
| Real map | 27.318 | 0.183 | 25.930 | 29.108 |
| Real map + FiLM4 | **27.408** | **0.177** | **26.044** | **29.148** |

The real map improves PSNR by 0.679 dB over fixed noise and 0.127 dB over energy-matched uniform noise, with lower LPIPS. FiLM4 adds 0.090 dB. Both High-U and Low-U fidelity improve, although the High-U difference between real and inverted maps is small. These controls support a contribution from spatial alignment beyond global noise magnitude, without attributing all gains to alignment.

**Component contributions.** Under the perceptual training recipe, adding the real map and FiLM4 raises PSNR from 26.649 to 27.290 dB and lowers LPIPS from 0.090 to 0.082. Adversarial polishing further lowers LPIPS to 0.076 while slightly reducing PSNR to 27.212 dB; post-fusion recovers fidelity to 28.998 dB with LPIPS 0.067. These progressive results separate the endpoint mechanism from perceptual sharpening and fidelity recovery; they are not a full factorial ablation (Appendix A).

**Causal hidden-token intervention.** With a fixed trained generator, input, prompt, timestep, and Gaussian noise, we compare real-map and energy-matched uniform conditions for both endpoint noise and FiLM. At layer 15, we transplant selected real-map hidden tokens into the uniform run. Correct High-U, Low-U, random, and spatially shifted swaps use matched token counts. This intervention probes the combined map-conditioned representation.

Table 3: Causal hidden-token intervention on the BurstFlicker-S dataset at DiT layer 15. High-U denotes the top 20% estimated-uncertainty image tokens. To-real L1 measures distance to the real-map output; localization is the High-U/Low-U output-change ratio. The real-map row is a reference pass.

| Intervention | High-U PSNR↑ | To-real L1↓ | Loc. ratio↑ |
|---|---|---|---|
| Uniform endpoint | 20.048 | 0.1670 | – |
| Random-token swap | 21.380 | 0.1304 | 7.01 |
| Low-U token swap | 20.154 | 0.1629 | 0.80 |
| Shifted High-U swap | 24.420 | 0.0622 | 13.83 |
| Correct High-U swap | **25.528** | **0.0349** | **15.03** |
| Real-map endpoint | 26.148 | 0.0000 | – |

Correct High-U replacement raises High-U PSNR from 20.048 to 25.528 dB, versus 21.380 for random and 20.154 for Low-U replacement. Shifting the High-U selection lowers the result to 24.420 dB. Thus the location of the transplanted states matters, and aligned High-U states recover much of the real-map behavior (26.148 dB). Complementary endpoint-only representation analysis holds FiLM uniform and finds stronger hidden responses in High-U regions; full protocols and layer analyses are in Appendix A.5.

## 5 Discussion and Limitations

Severe flicker removal requires local completion while preserving reliable content. Our evidence is limited to the BurstFlicker-S single-image setting and does not cover screen-recapture banding. The method uses an 8B-parameter prior, uncertainty estimation, and post-fusion refinement; under- or over-estimated uncertainty can respectively suppress needed completion or cause unnecessary edits.

## AI USE STATEMENT

We used generative AI tools for language polishing, code assistance, figure-layout prototyping, and consistency checks. We did not use generative AI tools to fabricate data, quantitative results, baselines, or references. All AI-assisted text, code, and visual materials were reviewed by the authors, and all reported numbers were produced by the evaluation scripts and experimental outputs described in the paper and appendix. We take responsibility for the content of this submission.

## REPRODUCIBILITY STATEMENT

We describe the dataset protocol, uncertainty-map construction, model components, training settings, evaluation metrics, and baseline protocol in the main paper and Appendix A. The appendix further reports implementation hyperparameters, stratified evaluation, representation analysis, and additional ablations to support reproducibility.

## REFERENCES

Qi Cai, Jingwen Chen, Chengmin Gao, Zijian Gong, Yehao Li, Yingwei Pan, Yi Peng, Zhaofan Qiu, Kai Yu, Yiheng Zhang, Hao Ai, Siying Bai, Yang Chen, Zhihui Chen, Fengbin Gao, Ying Guo, Dong Li, Zhen Shen, Leilei Shi, Jing Wang, Siyu Wang, Yimeng Wang, Rui Zheng, Ting Yao, and Tao Mei. HiDream-O1-Image: A natively unified image generative foundation model with pixel-level unified transformer, 2026.

Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *European Conference on Computer Vision*, 2022.

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*, 2022.

Yushun Fang, Yuxiang Chen, Shibo Yin, Qiang Hu, Jiangchao Yao, Ya Zhang, Xiaoyun Zhang, and Yanfeng Wang. One-step diffusion transformer for controllable real-world image super-resolution, 2025.

Zihao He, Shengchuan Zhang, Runze Hu, Yunhang Shen, and Yan Zhang. BUFF: Bayesian uncertainty guided diffusion probabilistic model for single image super-resolution, 2025.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Alexia Jolicoeur-Martineau. The relativistic discriminator: A key element missing from standard gan. In *International Conference on Learning Representations*, 2019.

Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. MUSIQ: Multi-scale image quality transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

Xiaodan Lin, Yangfu Li, Jianqing Zhu, and Huanqiang Zeng. Deflickercyclegan: Learning to detect and remove flickers in a single image. *IEEE Transactions on Image Processing*, 32:709–720, 2023. doi: 10.1109/TIP.2022.3231748.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Xiaoyang Liu, Zhengyan Zhou, Zihang Xu, Jiezhang Cao, Zheng Chen, and Yulun Zhang. FideDiff: Efficient diffusion model for high-fidelity image motion deblurring. In *International Conference on Learning Representations*, 2026.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023.

Qian Ning, Weisheng Dong, Xin Li, Jinjian Wu, and Guangming Shi. Uncertainty-driven loss for single image super-resolution. In *Advances in Neural Information Processing Systems*, 2021.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Lishen Qu, Zhihao Liu, Shihao Zhou, Yaqi Luo, Jie Liang, Hui Zeng, Lei Zhang, and Jufeng Yang. Burstdeflicker: A benchmark dataset for flicker removal in dynamic scenes. In *Advances in Neural Information Processing Systems*, 2025.

Lishen Qu, Zhihao Liu, Shihao Zhou, Jie Liang, Hui Zeng, Lei Zhang, and Jufeng Yang. It takes two: A duet of periodicity and directionality for burst flicker removal. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

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

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, et al. Wan: Open and advanced large-scale video generative models, 2025.

Jingkai Wang, Yixin Tang, Jue Gong, Jiatong Li, Li Shu, Libo Liu, Jianliang Lan, Yutong Liu, and Yulun Zhang. Spectral and trajectory regularization for diffusion transformer super-resolution, 2026.

Yufei Wang, Wenhan Yang, Xinyuan Chen, Yaohui Wang, Lanqing Guo, Lap-Pui Chau, Alex C. Kot, and Bihan Wen. Sinsr: Diffusion-based image super-resolution in a single step. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 25796–25805, 2024.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhengyi Wang, An Yang, et al. Qwen-Image technical report, 2025.

Rongyuan Wu, Lingchen Sun, Zhiyuan Ma, and Lei Zhang. One-step effective diffusion network for real-world image super-resolution, 2024.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2022.

Junjie Yin, Jiaju Li, and Hanfa Xing. QUSR: Quality-aware and uncertainty-guided image super-resolution diffusion model, 2026.

Zongsheng Yue, Jianyi Wang, and Chen Change Loy. Resshift: Efficient diffusion model for image super-resolution by residual shifting. In *Advances in Neural Information Processing Systems*, 2023.

Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5728–5739, 2022.

Leheng Zhang, Weiyi You, Kexuan Shi, and Shuhang Gu. Uncertainty-guided perturbation for image super-resolution diffusion model, 2025.

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

Libo Zhu, Zihan Zhou, Xiaoyang Liu, Weihang Zhang, Keyu Shi, Yifan Fu, and Yulun Zhang. RIFLE: Removal of image flicker-banding via latent diffusion enhancement, 2025.

Libo Zhu, Zihan Zhou, Zhiyi Zhou, Yiyang Qu, Weihang Zhang, Keyu Shi, Yifan Fu, and Yulun Zhang. CLEAR: Combined flicker-banding and moiré removal for screen-captured images, 2026.

# A Additional Experimental Details and Results

## A.1 Implementation Hyperparameters

Table 4: Additional implementation details of the selected FlickerDiff system. DP denotes data-parallel processes.

| Item | Setting |
|---|---|
| Generator | HiDream-O1 pixel-space generator with LoRA rank 32 on attention and MLP projections. |
| Endpoint | $t_p = 0.43$, $\sigma_0 = 8.0$, real raw uncertainty map with $\phi_{\max} = 0.4$, clamp $[0.2, 1.0]$. |
| FiLM4 | Raw uncertainty plus normalized coordinates, stride-32 map embedding, four zero-initialized FiLM residuals at evenly spaced denoising layers. |
| Coarse training | 21k steps, effective batch 2, bf16 mixed precision, learning rate $1 \times 10^{-4}$, $\lambda_1 = 0.5$, $\lambda_{\text{lpips}} = 1.0$. |
| GAN training | 20k base steps plus 10k polish, effective batch 2, generator/discriminator learning rates $1 \times 10^{-4}/5 \times 10^{-6}$ for base training and $3 \times 10^{-5}/2 \times 10^{-6}$ for polish, $\lambda_{\text{adv}} = 0.01$. |
| Discriminator | Frozen Wan VAE encoder, trainable Wan2.1-T2V-1.3B DiT with PatchGAN head, relativistic patch adversarial loss, one D update per G update. |
| Post-fusion | NAFNet-style direct refiner, width 64, middle block 1, encoder blocks $[1, 1, 1, 28]$, decoder blocks $[1, 1, 1, 1]$, 30k steps, batch 4, 512 patches. |
| Hardware | Main local experiments use RTX 4090 GPUs; coarse, FiLM, and GAN jobs use two DP processes; post-fusion is trained after cached generator inference. |

## A.2 Auxiliary Training and Refinement Details

One-step endpoint training with pixel reconstruction and LPIPS is stable and faithful, but it can average plausible high-frequency structures in severe flicker regions. This is especially visible around text strokes, thin edges, repeated patterns, and high-contrast display content. We therefore study adversarial supervision as a detail-oriented training signal for the coarse generator, following the observation in recent one-step DiT restoration that adversarial post-training can activate the generative prior and reduce over-smoothing (Fang et al., 2025). This component does not define the uncertainty mechanism; it is used to sharpen the uncertainty-guided Pixel-DiT output.

The discriminator is used only during training. The generator remains a pixel-space Pixel-DiT, while the discriminator judges RGB outputs through frozen Wan VAE features and a trainable Wan DiT with a lightweight PatchGAN head (Team Wan et al., 2025; Isola et al., 2017). We use a relativistic patch adversarial loss (Jolicoeur-Martineau, 2019) and discard the discriminator at inference. Detailed discriminator and polishing settings are given in Appendix A.1.

The coarse-generator objective is:

$$\mathcal{L}_G = \lambda_{\text{mse}}\mathcal{L}_{\text{endpoint}} + \lambda_1 \left\|\hat{x}_0 - I_{\text{GT}}\right\|_1 + \lambda_{\text{lpips}}\mathcal{L}_{\text{LPIPS}} + \lambda_{\text{adv}}\mathcal{L}_{\text{adv}}^G. \tag{18}$$

Unless otherwise specified, the reconstruction-trained coarse model uses $\lambda_{\text{mse}} = 1$, $\lambda_1 = 0.5$, $\lambda_{\text{lpips}} = 1$, and $\lambda_{\text{adv}} = 0$. The adversarial variant sets $\lambda_{\text{adv}} = 0.01$ with the same real-map endpoint policy and FiLM4 conditioning. Inference remains a single Pixel-DiT endpoint prediction, although the backbone itself is an 8B-parameter pretrained foundation model.

The adversarially enhanced coarse output improves perceptual quality but can slightly reduce local fidelity. For the deployed system, we therefore use a lightweight NAFNet-style post-fusion refiner (Chen et al., 2022). The refiner is trained after caching selected uncertainty-guided generator outputs, and takes the coarse output, the input image, a high-frequency image, and the uncertainty map as inputs:

$$I_{\text{out}} = R_\psi([I_{\text{coarse}}, I_{\text{LQ}}, H(I_{\text{LQ}}), B]), \tag{19}$$

and is trained with a direct RGB objective:

$$\mathcal{L}_{\text{fusion}} = 0.5\mathcal{L}_1 + 0.25\mathcal{L}_{\text{LPIPS}} + 0.25\mathcal{L}_{\text{grad}}. \tag{20}$$

The standard objective improves global fidelity but may overwrite useful generated details in highly uncertain regions. Our direct region-loss refiner therefore adds two uncertainty-aware consistency terms. We define a soft high-uncertainty mask

$$B_{\mathrm{hi}} = \mathrm{clip}\left(\frac{B-\tau}{1-\tau}, 0, 1\right), \quad B_{\mathrm{lo}} = 1 - B_{\mathrm{hi}}, \tag{21}$$

and optimize

$$\mathcal{L}_{\mathrm{fusion}}^{\mathrm{direct}} = \mathcal{L}_{\mathrm{fusion}} + \lambda_{\mathrm{keep}} \left\| B_{\mathrm{hi}} \odot (I_{\mathrm{out}} - I_{\mathrm{coarse}}) \right\|_1 + \lambda_{\mathrm{lo}} \left\| B_{\mathrm{lo}} \odot (I_{\mathrm{out}} - I_{\mathrm{GT}}) \right\|_1 . \tag{22}$$

The high-uncertainty term discourages the refiner from erasing coarse generated completions, while the low-uncertainty term strengthens fidelity in reliable regions. We use $\tau = 0.6$ and $\lambda_{\mathrm{keep}} = \lambda_{\mathrm{lo}} = 0.1$ for the direct post-fusion model. This stage is designed to recover fidelity and clean non-degraded areas; it is evaluated separately from the endpoint mechanism in the ablations.

### A.3 Inference Runtime

Table 5 reports a re-benchmarked practical runtime comparison on the BurstFlicker-S dataset. All methods are evaluated on $1024 \times 512$ crops with one visible RTX 4090 per inference process. We measure end-to-end script wall time per generated image, including image loading, method-side preprocessing, and PNG writing. Some local baseline scripts also compute lightweight PSNR/SSIM during inference; no additional PyIQA evaluation is included in the timing. For 50-step HiDream baselines, we run independent one-GPU shards and report the mean single-image latency across shards, rather than dividing total wall-clock time by the number of GPUs. The HiDream official zero-shot sampler is the original 50-step editing sampler, not our one-step endpoint adaptation. The adversarial discriminator is training-only, so the GAN-trained generator has the same inference graph as the coarse generator. The latency is the sum of one Pixel-DiT endpoint pass and the lightweight direct refiner.

Table 5: Inference runtime on the BurstFlicker-S dataset. Time is measured in seconds per $1024 \times 512$ image using end-to-end script wall time on RTX 4090 GPUs. Lower is better. The HiDream official zero-shot sampler is the original 50-step editing sampler, not the one-step endpoint adaptation used by FlickerDiff.

| Method | Steps | Time↓ |
|---|---|---|
| DeflickerCycleGAN | 1 | 0.45 |
| Restormer | 1 | 0.55 |
| PiSA-SR | 1 | 0.91 |
| OSEDiff | 1 | 0.54 |
| ResShift | 15 | 4.84 |
| SinSR[†] | 1 | 0.73 |
| StrSR-ZImage[‡] | 1 | 0.78 |
| HiDream multi-step | 50 | 12.57 |
| HiDream official zero-shot sampler | 50 | 23.40 |
| Ours coarse / GAN generator | 1 | 1.31 |
| Ours | 1+R | 1.79 |

[†]SinSR is measured with the locally available sf=1 10k checkpoint as a speed proxy because the selected main-table checkpoint is unavailable locally. [‡]StrSR-ZImage is computed from a completed full-test manifest timestamp because the selected checkpoint is unavailable locally.

Table 6: Uncertainty-stratified comparison on the BurstFlicker-S dataset. Images are sorted by the top-10% mean value of the estimated uncertainty map and split into equal mild, medium, and severe subsets. Metrics are computed on full images within each subset. This table complements Table 1 by testing whether each method remains reliable as uncertainty increases.

| Method | PSNR↑ | | | LPIPS↓ | | | DISTS↓ | | |
|---|---|---|---|---|---|---|---|---|---|
| | Mild | Medium | Severe | Mild | Medium | Severe | Mild | Medium | Severe |
| Restormer | 28.285 | 25.484 | 21.819 | 0.0668 | 0.1324 | 0.1891 | 0.0570 | 0.1035 | 0.1303 |
| PiSA-SR | 26.705 | 25.329 | 22.074 | 0.1105 | 0.1374 | 0.1818 | 0.0909 | 0.1145 | 0.1333 |
| OSEDiff | 26.442 | 26.344 | 24.419 | 0.0752 | 0.1020 | 0.1357 | 0.0560 | 0.0738 | 0.0857 |
| ResShift | 28.635 | 26.973 | 24.859 | 0.0779 | 0.1186 | 0.1578 | 0.0672 | 0.0937 | 0.1091 |
| StrSR-ZImage | 27.063 | 25.153 | 22.145 | 0.0727 | 0.1223 | 0.1679 | 0.0560 | 0.0895 | 0.1057 |
| Ours coarse | 27.733 | 28.321 | 25.818 | 0.0666 | 0.0818 | 0.0972 | 0.0805 | 0.0865 | 0.0936 |
| Ours | **30.133** | **29.667** | **27.202** | **0.0465** | **0.0694** | **0.0840** | **0.0471** | **0.0676** | **0.0725** |

## A.4 Map Reliability Visualization

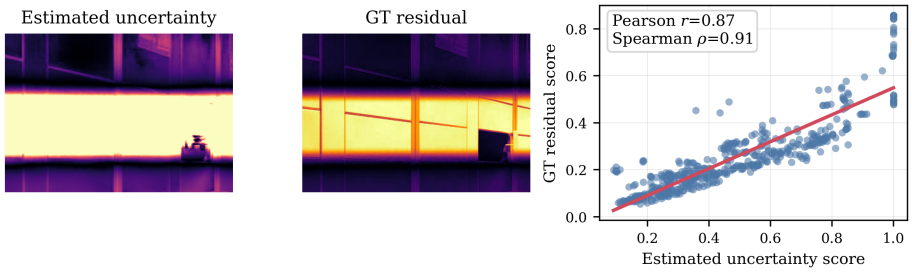


Figure 4: Reliability analysis of the estimated uncertainty map. A representative crop shows that high uncertainty aligns with large ground-truth residuals, and the test-set scatter plot confirms a strong correlation between uncertainty and residual scores. The ground-truth residual is used only for analysis, not for map construction, training, or inference.

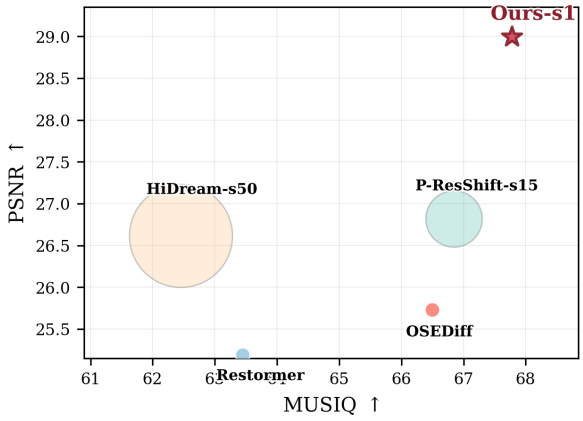


Figure 5: Fidelity–perception trade-off of representative methods on the BurstFlicker-S dataset. PSNR measures paired fidelity and MUSIQ measures no-reference perceptual quality. Bubble area is proportional to the number of inference steps.

Table 6 shows that estimated uncertainty is a meaningful difficulty indicator: most baselines degrade substantially from mild to severe subsets. The model gives the best full-reference scores in all three subsets, suggesting that uncertainty-guided generation and fidelity recovery remain effective as local content becomes less reliable. The severe subset is especially diagnostic for the proposed mechanism. The coarse one-step generator without post-fusion already obtains 25.818 dB PSNR

Table 7: Region-level evaluation on the BurstFlicker-S dataset. High-U denotes the top 20% estimated-uncertainty pixels and Low-U denotes the bottom 50% pixels.

| Method | High-U PSNR↑ | High-U MAE↓ | Low-U PSNR↑ | Low-U MAE↓ |
|---|---|---|---|---|
| Restormer | 17.935 | 0.0775 | 27.051 | 0.0247 |
| OSEDiff | 21.064 | 0.0514 | 26.522 | 0.0261 |
| ResShift | 20.858 | 0.0493 | <u>27.230</u> | <u>0.0226</u> |
| Ours coarse | <u>22.748</u> | <u>0.0378</u> | 26.799 | <u>0.0226</u> |
| Ours | **23.284** | **0.0347** | **27.954** | **0.0188** |

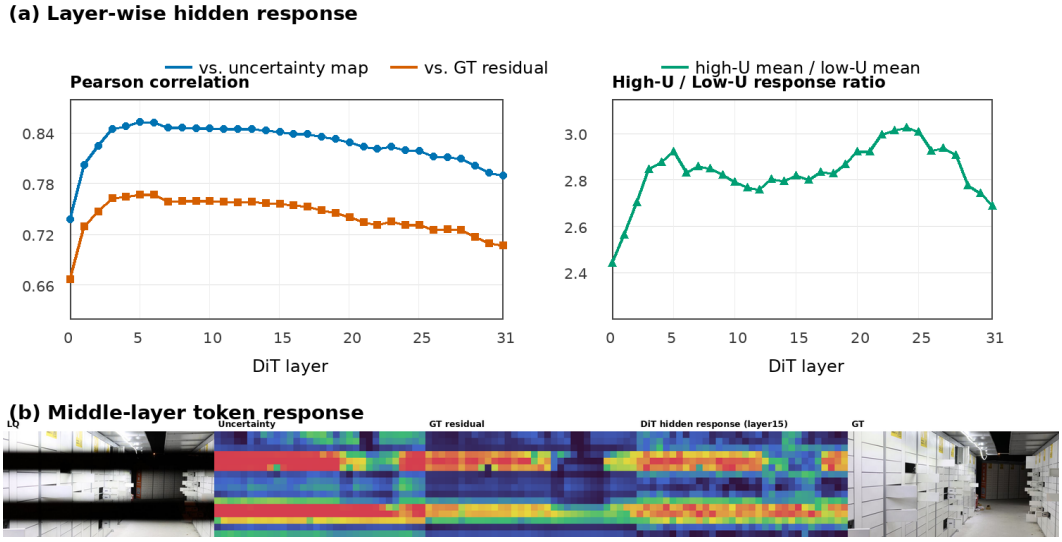


Figure 6: Representation analysis of uncertainty-modulated endpoint noise on the BurstFlicker-S dataset. (a) Across all 32 DiT layers, hidden-token displacement remains correlated with estimated uncertainty and ground-truth residual magnitude. (b) A representative middle-layer example shows that the hidden response aligns with high-uncertainty flicker bands and the corresponding ground-truth residual.

and 0.0972 LPIPS on severe samples, ahead of the strongest baseline scores in this table, indicating that difficult-region recovery is not introduced only by the refiner.

Table 7 further verifies the spatial behavior. The coarse one-step generator already improves High-U PSNR over representative baselines, while the refiner improves both High-U recovery and Low-U fidelity.

### A.5 Representation Analysis

To verify that the uncertainty map affects the DiT computation rather than only changing output statistics, we analyze hidden-token responses under a controlled counterfactual. For each test image, we run the same trained generator with the same random seed and prompt under two endpoints: the proposed spatial endpoint noise $\sigma(p) = \sigma_0 g(B_p)$ and an energy-matched uniform endpoint whose RMS noise magnitude equals that of the spatial map. The FiLM condition is kept spatially uniform in both forwards, so the measured difference isolates the effect of uncertainty-modulated endpoint noise. For layer $\ell$, we compute the image-token displacement

$$D_\ell(i) = \left\| h_\ell^{\text{map}}(i) - h_\ell^{\text{uni}}(i) \right\|_2 , \tag{23}$$

where $i$ indexes the $32 \times 16$ image-token grid for a $1024 \times 512$ input. We then compare $D_\ell$ with the downsampled uncertainty map and the downsampled ground-truth residual.

Figure 6 shows that endpoint modulation induces spatially aligned changes throughout the DiT. The two curves in the left plot measure how strongly the layer-wise hidden response correlates with the

Table 8: Completion–preservation policy alignment on the BurstFlicker-S dataset. Output alignment measures whether the model edits where the paired GT differs from the input. Hidden alignment measures whether the hidden displacement from the uniform-energy run predicts the top-correction tokens.

| View | Condition | Corr.↑ | High-U error↓ / AUC↑ | Low-U edit↓ / Ratio↑ | Low-U error↓ |
|---|---|---|---|---|---|
| Output | Inverted | -0.227 | 0.385 | 0.403 | 0.382 |
| | Random | 0.514 | 0.210 | 0.080 | 0.075 |
| | Uniform | 0.707 | 0.184 | 0.075 | 0.055 |
| | Real map | **0.902** | **0.074** | **0.056** | **0.046** |
| Hidden L15 | Inverted | -0.615 | 0.134 | 0.72 | – |
| | Random | -0.017 | 0.488 | 1.01 | – |
| | Real map | **0.731** | **0.814** | **2.36** | – |
| Hidden L23 | Inverted | -0.561 | 0.157 | 0.76 | – |
| | Random | -0.048 | 0.473 | 0.99 | – |
| | Real map | **0.708** | **0.815** | **2.34** | – |

uncertainty map and with the true residual, respectively. The right plot measures localization: it divides the average response on high-uncertainty tokens by the average response on low-uncertainty tokens. Averaged over all layers, the hidden response has Pearson correlation $0.827$ with the estimated uncertainty map and $0.743$ with the ground-truth residual. The high-uncertainty response is consistently about $2.4$–$3.0\times$ larger than the low-uncertainty response. This supports the mechanism in Sec. 3: the uncertainty map does not merely change the global noise level, but spatially redirects the one-step flow representation toward unreliable regions.

**Causal token intervention.** Correlation alone does not prove that the map-induced hidden states are used by the generator. We therefore perform a causal intervention with the same trained coarse generator. For each test image, we run two forwards with identical LQ input, prompt, timestep, and sampled Gaussian noise. One forward uses the real uncertainty condition for both endpoint sigma and FiLM, while the counterfactual forward uses an energy-matched uniform condition. At DiT layer $\ell$, we replace selected image-token hidden states in the uniform run by their counterparts from the real-map run and continue the forward pass. Correct High-U swap replaces the top 20% estimated-uncertainty tokens; Low-U, Random, and Shifted High-U swaps are controls with the same number of replaced tokens where applicable. We report High-U PSNR against the paired GT, To-real L1 to the real-map output, and a localization ratio measuring high-uncertainty output change divided by low-uncertainty output change.

The intervention results are reported in Table 3 in the main text.

Table 3 shows that spatially aligned High-U token replacement recovers most of the real-map behavior, raising High-U PSNR from 20.05 dB to 25.53 dB at the middle DiT layer. In contrast, random-token and Low-U swaps give only 21.38 dB and 20.15 dB, respectively. Shifted High-U swap is substantially stronger than random replacement but weaker than the correctly aligned High-U swap, indicating that the effect is not explained only by the number of modified tokens or global noise energy. This complements Fig. 6: endpoint modulation creates spatially aligned hidden responses, and the token intervention shows that the corresponding high-uncertainty token states causally affect local restoration.

**Completion–preservation policy alignment.** Since the generator is trained with real uncertainty maps, replacing the real map at inference does not test an unseen conditioning distribution. Instead, it probes whether the learned generator actually uses the spatially aligned map to activate the intended restoration policy. For each token, we define the ground-truth correction magnitude $C_{\mathrm{gt}} = |I_{\mathrm{GT}} - I_{\mathrm{LQ}}|$, the output edit magnitude $C_{\mathrm{out}} = |I_{\mathrm{out}} - I_{\mathrm{LQ}}|$, and the restoration error $E_{\mathrm{out}} = |I_{\mathrm{out}} - I_{\mathrm{GT}}|$, all averaged and downsampled to the image-token grid. We then compare real, uniform-energy, random, and inverted map conditions using the same trained model and the same sampled Gaussian noise.

Table 8 shows that the real map condition makes the output edits better aligned with the GT correction magnitude than uniform, random, or inverted conditions, while also reducing edits and errors

Table 9: Attention-routing enrichment on the BurstFlicker-S dataset. Values larger than one indicate that the target key group receives more attention than expected from its token fraction.

| Routing | Uniform | Inverted | Real map |
|---|---|---|---|
| High-U → Boundary, L15 | 1.01 | 0.92 | **1.09** |
| High-U → Boundary, L23 | 1.09 | 1.05 | **1.19** |
| Low-U → Low-U, L15 | 1.56 | 1.46 | **1.60** |
| Low-U → Low-U, L23 | 1.72 | 1.64 | **1.78** |

in low-uncertainty regions. At the representation level, the hidden displacement induced by the real map strongly predicts top-correction tokens, with AUC above 0.81 at layers 15 and 23, whereas random maps are near chance and inverted maps are anti-correlated. This indicates that the trained generator is not map-agnostic: the spatially aligned map condition activates a completion–preservation policy that matches where the paired target requires large or small changes.

**Attention routing.** As a complementary analysis, we also measure attention enrichment between uncertainty-defined image-token groups. We focus on two routing patterns directly related to the proposed behavior: High-U queries attending to boundary tokens, which can provide nearby reliable context for local completion, and Low-U queries attending to Low-U tokens, which reflects conservative routing in reliable regions. Attention weights are not used as causal evidence; they only provide an additional view of how the DiT computation changes under different map conditions.

Table 9 shows that the real uncertainty map mildly but consistently increases boundary-context routing for high-uncertainty queries and low-region routing for low-uncertainty queries in the middle-to-late DiT layers. This trend is consistent with the intended completion–fidelity behavior, while the causal token intervention above remains the stronger evidence that these region-specific hidden states affect the restoration.

**Real-image wrong-map inference probe.** To further test whether the coarse generator relies on the spatial alignment of the uncertainty map rather than only on the pretrained HiDream prior, we run an inference-only mismatch probe on the BurstFlicker-S dataset. The generator checkpoint, prompt, timestep, sampled Gaussian noise, endpoint parameters, and FiLM4 configuration are fixed. For each input image, the aligned-map run uses its own estimated uncertainty map, while the wrong-map run replaces it with a real uncertainty map from another test image. Thus both runs use in-distribution real maps, but only the wrong-map run breaks image–map alignment.

Table 10: Real-image wrong-map inference probe on the BurstFlicker-S dataset. Both rows use the same trained FiLM4 coarse generator and identical random seeds; the wrong-map row replaces each image's aligned estimated uncertainty map with a real map from another test image. High-U and Low-U regions are computed from the correct estimated map of each image.

| Condition | PSNR↑ | SSIM↑ | LPIPS↓ | DISTS↓ | High-U PSNR↑ | High-U MAE↓ | Low-U PSNR↑ | Low-U MAE↓ |
|---|---|---|---|---|---|---|---|---|
| Aligned real map | **27.290** | **0.8676** | **0.0819** | **0.0869** | **22.748** | **0.0378** | **26.799** | **0.0226** |
| Wrong real map | 21.009 | 0.8157 | 0.1453 | 0.1163 | 13.612 | 0.1294 | 22.399 | 0.0404 |

Table 10 shows a large degradation when the map is replaced by another real uncertainty map at inference. This control is different from random or inverted maps: the wrong maps have realistic estimated-map statistics, but their high-uncertainty regions do not correspond to the current input. The result supports the intended interpretation that the map acts as a spatial reliability controller. The model prior is important for completion, but it must be activated at the right locations to improve high-uncertainty recovery while preserving low-uncertainty content.

**Synthetic flicker-only alignment probe.** The estimated uncertainty map is not a binary flicker mask: it also responds to text strokes, object boundaries, attenuated textures, and other high-frequency Restormer–input disagreements. To disentangle band alignment from these natural-image responses, we build an inference-only synthetic probe using flicker-only uncertainty fields. For each clean test image, we synthesize a soft horizontal band field $M_{\mathrm{syn}} \in [0, 1]$ with mild curvature, low-

frequency envelope variation, and sparsity statistics matched to the real $\phi = 0.40$ maps. We then degrade the clean image by multiplicative under-exposure,

$$I_{\text{deg}} = I_{\text{GT}} \odot (1 - 0.65\, M_{\text{syn}}), \tag{24}$$

and run the same trained FiLM4 coarse generator using either the aligned synthetic map or map controls. Uniform RMS uses a spatially constant map with the same RMS value, while shifted, inverted, random-smooth, and wrong-stripe maps preserve different aspects of the map statistics but break spatial alignment.

Table 11: Synthetic flicker-only spatial alignment probe on 80 shuffled test images. This is an inference-only controllability analysis, not a real-flicker benchmark. High-D and Low-D denote the top 20% and bottom 50% pixels of the synthetic degradation field. High-D reduction is the decrease in High-D MAE relative to the degraded input.

| Condition | PSNR↑ | High-D MAE↓ | High-D red.↑ | Low-D MAE↓ | Edit loc.↑ |
|---|---|---|---|---|---|
| Degraded input | 23.973 | 0.1165 | 0.0000 | 0.0046 | – |
| Aligned map | **26.375** | **0.0601** | **0.0564** | 0.0234 | **6.93** |
| Uniform RMS | 24.067 | <u>0.0711</u> | <u>0.0455</u> | 0.0423 | 1.71 |
| Shifted map | 21.365 | 0.0971 | 0.0194 | 0.0516 | 1.02 |
| Inverted map | 20.388 | 0.1077 | 0.0088 | 0.0659 | 0.42 |
| Random smooth | 22.564 | 0.0743 | 0.0423 | 0.0522 | 1.74 |
| Wrong stripe | 20.663 | 0.0904 | 0.0261 | 0.0494 | 1.91 |

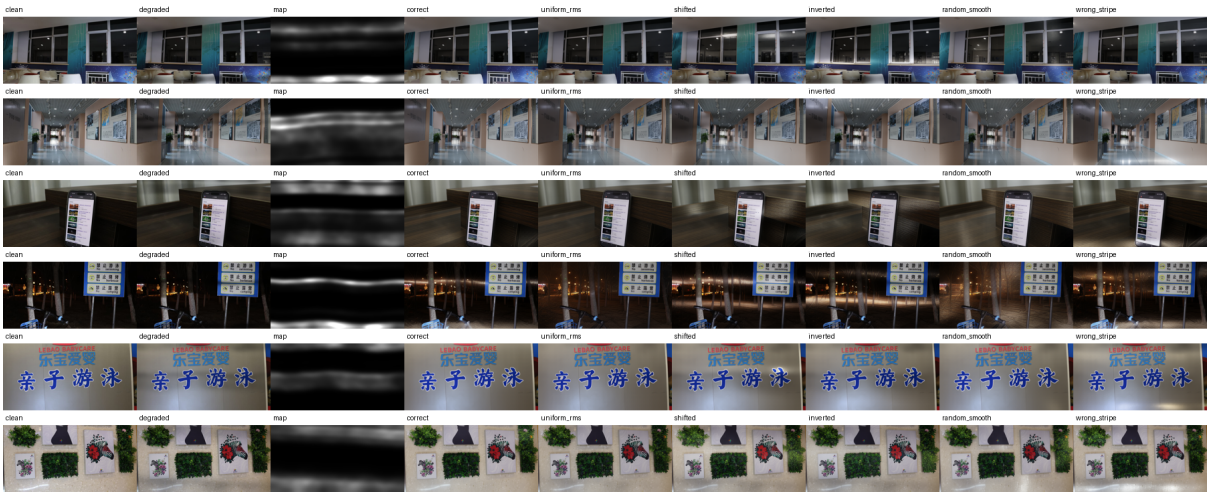


Figure 7: Qualitative examples from the synthetic flicker-only alignment probe. The aligned map restores the under-exposed bands more locally, whereas shifted, inverted, and wrong-stripe controls introduce edits at mismatched spatial locations.

Table 11 and Fig. 7 show that the aligned synthetic map produces the strongest High-D recovery and the largest edit-localization ratio. Compared with the RMS-matched uniform map, it also causes substantially less Low-D error, indicating that the effect is not explained by global endpoint noise energy alone. Because this synthetic degradation preserves most image structure, the degraded input can retain favorable perceptual scores; we therefore use this probe only as spatial-alignment evidence for the endpoint controller.

**Zero-shot HiDream-O1 prior.** We also evaluate the original HiDream-O1 model without BurstFlicker-S fine-tuning to test whether the pretrained image prior alone can remove single-image flicker artifacts. This diagnostic is not used as a main-table baseline because the official zero-shot editing pipeline follows a keep-aspect resizing protocol rather than the task-adapted $1024 \times 512$

training protocol. We therefore run the official zero-shot setting, then center-crop both restored outputs and paired ground-truth images to $1024 \times 512$ for metric computation.

Table 12: Zero-shot HiDream-O1 prior on the BurstFlicker-S dataset. The model is used without task-specific fine-tuning. Outputs and GT images are center-cropped to $1024 \times 512$ for PyIQA metric computation.

| Method | PSNR↑ | SSIM↑ | LPIPS↓ | DISTS↓ | MUSIQ↑ | MANIQA↑ |
|---|---|---|---|---|---|---|
| HiDream-O1 zero-shot | 19.460 | 0.7513 | 0.2603 | 0.1575 | 61.358 | 0.3528 |
| Ours | **28.998** | **0.9140** | **0.0670** | **0.0620** | **67.776** | **0.4370** |

Table 12 shows that the unadapted HiDream-O1 prior is insufficient for this paired flicker-removal setting. The result should be interpreted as a foundation-prior diagnostic rather than a protocol-matched restoration baseline: without uncertainty-guided endpoint modulation and BurstFlicker-S adaptation, the pretrained model does not reliably preserve paired image content or remove structured flicker artifacts.

We further visualize the middle-layer token space in Fig. 8. Using the same trained generator, high-uncertainty and low-uncertainty tokens are separated by PCA with a nearest-centroid accuracy of $80.1\%$ in the projected space. This analysis is not used as a performance metric; rather, it provides output-independent evidence that the uncertainty signal is reflected in the DiT hidden representation.

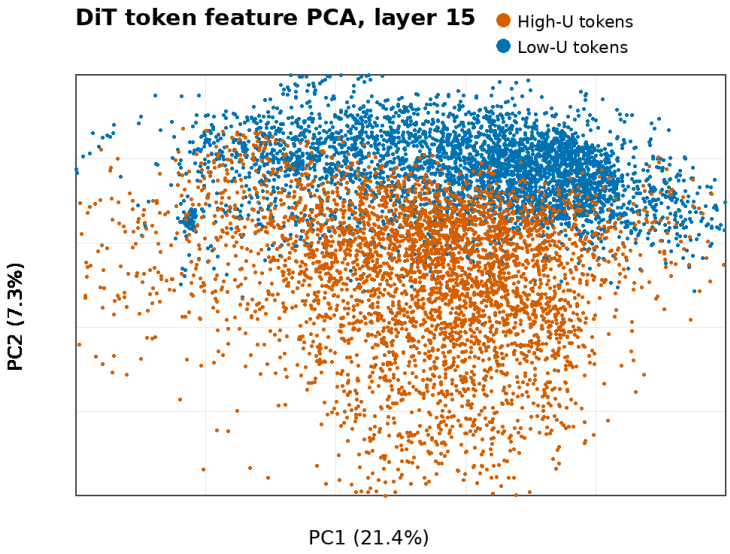


Figure 8: PCA visualization of DiT hidden image-token features at layer 15 on the BurstFlicker-S dataset. Tokens are sampled from the top 20% and bottom 20% estimated-uncertainty regions, with per-image feature normalization before PCA to reduce image-identity bias. High- and low-uncertainty tokens form separable clusters, suggesting that the uncertainty-guided endpoint also organizes the middle-layer token representation.

### A.6 Additional Ablations

Table 14 localizes the endpoint-control effect using the same controlled protocol. The real-map variants give the strongest High-U and Low-U PSNR when regions are defined by the real uncertainty map. This indicates that spatial endpoint modulation is not only a full-image gain: aligned uncertainty improves recovery in unreliable regions while retaining the best fidelity in reliable regions. The differences in High-U error metrics are smaller than the full-image trends, so we use this table as a region-level sanity check rather than as a separate primary-performance claim.

Table 16 verifies that the map remains useful after moving from the mechanism protocol to stronger one-step generator settings. For the reconstruction-trained coarse model, real-map endpoint modulation improves PSNR by 0.62 dB, reduces LPIPS from 0.090 to 0.083, and reduces DISTS from 0.125 to 0.087. Adding FiLM4 gives a further but smaller improvement, indicating that the main gain comes from endpoint modulation while explicit map conditioning is a useful refinement. The resulting coarse generator is not merely a weak intermediate output: compared with the external baselines in Table 1, it gives the best PSNR and LPIPS among non-refined methods in this view, while leaving room for stronger SSIM from conservative restoration and stronger DISTS from OSEDiff. For adversarial training, the real uncertainty map also improves the no-map GAN across the reported metrics. Adding FiLM4 and adversarial polishing further improves LPIPS, DISTS, and no-reference perceptual scores over the endpoint-only GAN, supporting its use as the selected perceptual generator variant.

**Selective gated-FiLM analysis.** We additionally test whether the FiLM intervention itself can be made more selective. Starting from the selected coarse-generator recipe, we replace the ordinary FiLM residual $h_\ell \leftarrow h_\ell + r_\ell \delta_\ell$ with a token-scalar gated form

$$h_\ell \leftarrow h_\ell + r_\ell \, q_\ell \, \delta_\ell, \qquad q_\ell = \sigma(W_\ell z_\ell + b_\ell), \tag{25}$$

where $q_\ell \in \mathbb{R}^{N\times 1}$ is predicted at the same four FiLM layers. We compare a map-only gate, where $z_\ell$ is the tokenized uncertainty map, and a map+hidden gate, where $z_\ell$ additionally includes a detached hidden-token summary. For the regularized variants, we penalize low-uncertainty activation with $\lambda_{\text{gate}}\mathbb{E}_\ell[(1 - B_{\text{tok}})q_\ell]$ and set $\lambda_{\text{gate}} = 5 \times 10^{-4}$. This ablation changes only the coarse-generator

Table 13: Controlled endpoint-mechanism ablation under pure reconstruction training. All variants use the same HiDream-O1 backbone, LoRA rank, training budget, crop size, and reconstruction/endpoint losses, without LPIPS loss, adversarial supervision, or post-fusion. Only the endpoint map and optional FiLM conditioning are changed. This table isolates the effect of spatial endpoint modulation rather than maximizing perceptual quality; the selected LPIPS-trained coarse generator is reported in Table 16. Uniform energy uses a spatially constant map with the same global RMS noise energy as the real map.

| Method | PSNR↑ | SSIM↑ | LPIPS↓ | DISTS↓ | MUSIQ↑ | MANIQA↑ |
|---|---|---|---|---|---|---|
| Fixed $t_p = 0.43$ | 26.639 | 0.852 | 0.207 | 0.164 | 53.293 | 0.313 |
| Uniform energy | 27.191 | 0.870 | 0.192 | 0.154 | 55.289 | 0.330 |
| Random smooth | 26.703 | 0.862 | 0.209 | 0.164 | 52.456 | 0.317 |
| Inverted map | 27.074 | 0.862 | 0.192 | 0.153 | 54.961 | 0.323 |
| Real uncertainty map | <u>27.318</u> | <u>0.874</u> | <u>0.183</u> | <u>0.152</u> | <u>56.989</u> | <u>0.341</u> |
| Real map + FiLM4 | **27.408** | **0.876** | **0.177** | **0.148** | **57.578** | **0.344** |

Table 14: Region-level view of the controlled endpoint ablation. All variants are the same pure-reconstruction models as Table 13. High-U and Low-U regions are always defined by the real uncertainty map, independent of the endpoint map used by each variant. High-U denotes the top 20% pixels and Low-U denotes the bottom 50% pixels.

| Endpoint map | Full PSNR↑ | High-U PSNR↑ | Low-U PSNR↑ |
|---|---|---|---|
| Fixed $t_p = 0.43$ | 26.639 | 25.557 | 28.168 |
| Uniform energy | 27.191 | 25.825 | 28.950 |
| Random smooth | 26.703 | 25.539 | 28.351 |
| Inverted map | 27.074 | 25.924 | 28.695 |
| Real uncertainty map | <u>27.318</u> | <u>25.930</u> | <u>29.108</u> |
| Real map + FiLM4 | **27.408** | **26.044** | **29.148** |

FiLM conditioning; endpoint modulation, estimated maps, training budget, and losses remain unchanged, and the adversarial/post-fusion stages are not retrained on these gated variants.

Table 17 shows a small but consistent gain from the map+hidden gated variant over the ordinary FiLM4 coarse generator. The improvement is modest in full-image metrics, but the gate statistics in Table 18 show the intended selective behavior: the learned gates activate more strongly on high-uncertainty tokens than on low-uncertainty tokens, especially in the later FiLM stages. Map-only gates are less selective and tend to stay globally active. We therefore treat gated-FiLM as supporting evidence that uncertainty-conditioned hidden-token intervention can be learned selectively, rather than as the deployed recipe.

### A.7 Refiner Efficiency–Quality Trade-off

The system uses the full direct post-fusion refiner for the main results. To characterize the deployment trade-off without changing the generator, we also evaluate smaller residual refiners using the same cached coarse/GAN outputs. The width-32 residual-lite8 model is trained for 10k steps, while the width-48 variant is trained for 15k steps; these are exploratory capacity comparisons rather than a replacement for the main training recipe.

Table 19: Efficiency–quality trade-off for post-fusion refiners using fixed cached generator outputs. The refiner does not change the coarse generator or the main result. Runtime is measured per $1024 \times 512$ image on an RTX 4090.

| Refiner | Params (M) | PSNR↑ | LPIPS↓ | MUSIQ↑ | High-U PSNR↑ | Low-U PSNR↑ | Time↓ |
|---|---|---|---|---|---|---|---|
| Full direct | 67.9 | **28.998** | **0.0667** | **67.776** | 23.284 | **27.954** | 0.0849 |
| Residual-lite8, w32 | 7.8 | 28.753 | 0.0694 | 65.749 | 23.311 | 27.865 | **0.0275** |
| Residual-lite8, w48 | 17.4 | 28.910 | 0.0664 | 67.254 | **23.328** | 27.948 | 0.0409 |

Table 15: Progressive component ablation on the BurstFlicker-S dataset. The columns follow the construction order of the system rather than a full factorial design: Base removes uncertainty-map guidance, adversarial supervision, and post-fusion; +Map uses the selected real-map endpoint policy with FiLM4 conditioning; +GAN adds adversarial detail supervision; Ours further adds the direct region-loss post-fusion refiner.

| Metric | Base | +Map | +GAN | Ours (+Post-fusion) |
|---|---|---|---|---|
| PSNR↑ | 26.649 | 27.290 | 27.212 | **28.998** |
| SSIM↑ | 0.841 | 0.868 | 0.863 | **0.914** |
| LPIPS↓ | 0.090 | 0.082 | 0.076 | **0.067** |
| DISTS↓ | 0.125 | 0.087 | 0.078 | **0.062** |
| MUSIQ↑ | 62.312 | 64.415 | 66.413 | **67.776** |
| MANIQA↑ | 0.344 | 0.368 | 0.403 | **0.437** |

Table 16: Effect of the uncertainty map under selected one-step generator training configurations on the BurstFlicker-S dataset. The real-map variants use the raw uncertainty map to modulate endpoint noise; the no-map variants keep the corresponding architecture and loss but remove spatial map guidance.

| Variant | PSNR↑ | SSIM↑ | LPIPS↓ | DISTS↓ | MUSIQ↑ | MANIQA↑ |
|---|---|---|---|---|---|---|
| Coarse, no map | 26.649 | 0.841 | 0.090 | 0.125 | 62.312 | 0.344 |
| Coarse, real map | <u>27.266</u> | <u>0.867</u> | 0.083 | 0.087 | 64.323 | 0.368 |
| Coarse, real map + FiLM4 | **27.290** | **0.868** | 0.082 | 0.087 | 64.415 | 0.368 |
| GAN, no map | 26.816 | 0.840 | 0.082 | 0.117 | 64.844 | 0.368 |
| GAN, real map | 27.186 | 0.859 | <u>0.081</u> | <u>0.082</u> | <u>65.889</u> | <u>0.402</u> |
| GAN, real map + FiLM4 | 27.212 | 0.863 | **0.076** | **0.078** | **66.413** | **0.403** |

The width-32 residual-lite8 refiner is about $3.1\times$ faster and reduces refiner parameters by roughly $88\%$, while remaining within $0.25$ dB PSNR and $0.003$ LPIPS of the full refiner. Its no-reference perceptual score is lower than the selected external baselines, so we do not use it as the deployed variant. The width-48 model is a stronger efficiency candidate, but it uses a different short training schedule and is therefore reported only as an appendix trade-off study.

### A.8 Output-Level Banding Diagnostics

We additionally apply the row-wise robust-profile and one-dimensional FFT diagnostics described in the accompanying diagnostic tool to the outputs associated with the current main-table methods. These quantities are output-level signals rather than ground-truth restoration metrics. The severe subset is selected by the input's 100 largest output-independent periodic peaks, not by the output recoverability label and not by the High-U pixel mask. The R–B profile correlation is only a heuristic measure of channel-wise synchronization, while clipping and recoverability labels should be interpreted as failure-analysis cues rather than evidence that lost information has been recovered.

Table 17: Gated-FiLM ablation on the LPIPS-trained coarse generator. Full-image metrics are computed with the same pyiqa protocol as Table 1; High-U and Low-U MAE use estimated-uncertainty masks. Gated variants are analysis models and are not used to replace the selected GAN/refiner pipeline.

| Variant | PSNR↑ | SSIM↑ | LPIPS↓ | DISTS↓ | High-U MAE↓ | Low-U MAE↓ |
|---|---|---|---|---|---|---|
| FiLM4 coarse | 27.290 | 0.868 | 0.082 | 0.087 | 0.0378 | 0.0226 |
| Map gate, $\lambda = 0$ | 27.332 | 0.869 | 0.081 | 0.086 | 0.0377 | 0.0227 |
| Map gate, $\lambda = 5\times10^{-4}$ | 27.293 | 0.867 | 0.083 | 0.088 | 0.0384 | 0.0228 |
| Map+hidden gate, $\lambda = 0$ | 27.322 | 0.868 | 0.082 | 0.087 | 0.0379 | 0.0228 |
| Map+hidden gate, $\lambda = 5\times10^{-4}$ | **27.381** | **0.870** | **0.081** | **0.086** | **0.0373** | **0.0226** |

Table 18: Gate statistics for the map+hidden gated-FiLM variant with $\lambda_{\text{gate}} = 5\times10^{-4}$ on the BurstFlicker-S dataset. High-U and Low-U are the top 20% and bottom 50% estimated-uncertainty tokens per image.

| Stage | Mean gate | High-U | Low-U | High/Low | Active $> 0.5$ | Corr. w/ U |
|---|---|---|---|---|---|---|
| 0 | 0.759 | 0.772 | 0.756 | 1.02 | 1.00 | 0.161 |
| 1 | 0.516 | 0.660 | 0.467 | 1.41 | 0.32 | 0.660 |
| 2 | 0.866 | 0.941 | 0.828 | 1.14 | 1.00 | 0.606 |
| 3 | 0.456 | 0.595 | 0.393 | 1.52 | 0.26 | 0.633 |

Table 20: Supplementary output-level banding diagnostics on the BurstFlicker-S dataset. Overall peak is the median FFT peak strength over all outputs. Severe peak is evaluated on the most strongly banded inputs. These diagnostics are not a replacement for the paired metrics in Tables 1 and 7.

| Method | Overall peak↓ | Severe peak↓ | Amp.↓ | R–B corr.↑ | Dark clip↓ | Bright clip↓ | Mostly rec.↑ |
|---|---|---|---|---|---|---|---|
| Aligned input | 1029.1 | 8388.3 | 0.0964 | 0.9111 | 0.000194 | 0.001597 | 83.8% |
| DeflickerCycleGAN | 433.6 | 348.9 | 0.0644 | 0.8287 | 0.000023 | 0.000450 | 92.4% |
| Restormer | 459.0 | 495.6 | 0.0563 | 0.8714 | 0.000075 | **0.000142** | 94.4% |
| PiSA-SR | 511.1 | 358.1 | 0.0583 | 0.8730 | 0.005491 | 0.000800 | 71.2% |
| OSEDiff | 419.9 | 306.3 | 0.0521 | 0.8694 | 0.000053 | 0.000895 | 91.2% |
| ResShift | 405.5 | 289.2 | 0.0550 | 0.8758 | 0.000042 | 0.000381 | 91.4% |
| SinSR | 607.8 | 528.0 | **0.0459** | 0.8581 | **0.000000** | 0.000171 | 92.0% |
| StrSR-ZImage | 507.3 | 684.1 | 0.0570 | 0.8915 | 0.000046 | 0.002542 | 89.8% |
| HiDream multi-step | 442.6 | 281.0 | 0.0504 | 0.8769 | 0.000063 | 0.000731 | 86.2% |
| Ours coarse | 442.1 | 276.9 | 0.0489 | 0.8716 | 0.000099 | 0.000444 | 89.2% |
| Ours | **374.0** | **259.1** | 0.0523 | 0.8734 | 0.000046 | 0.000699 | 88.8% |

The diagnostics show a complementary and non-uniform pattern. Ours has the lowest overall and severe periodic peak among the listed outputs, while Ours coarse and SinSR have lower median amplitude ratios. This means that peak suppression and amplitude flattening are not equivalent, and either can be affected by smoothing or content changes. The output also has lower R–B profile correlation than the aligned input, so this table does not support a claim of uniformly improved color synchronization. We therefore use these measurements only to identify residual banding and possible failure cases; the primary conclusions remain based on paired full-image and uncertainty-stratified metrics.

### A.9 Qualitative Analysis

Qualitative inspection on representative severe flicker cases shows the main behavior of the proposed method. Conservative baselines often remove visible banding but leave severe regions incomplete, whereas our model recovers more plausible local structure and the post-fusion stage improves consistency with the input. A remaining failure mode is over-smoothing of some generated details, motivating refiners that preserve high-uncertainty completions more selectively.

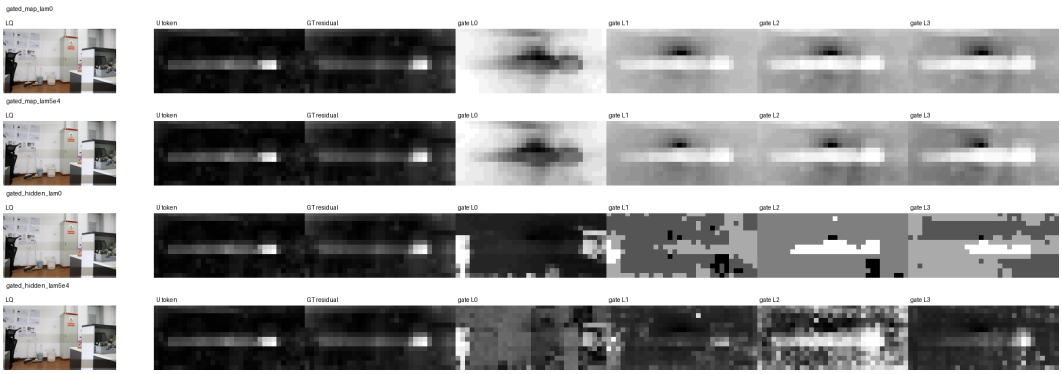


Figure 9: Example token-gate heatmaps for gated-FiLM variants. Map-only gates tend to remain broadly active, while the map+hidden gate with low-uncertainty sparsity produces more selective activations that better concentrate around high-uncertainty and high-residual regions.

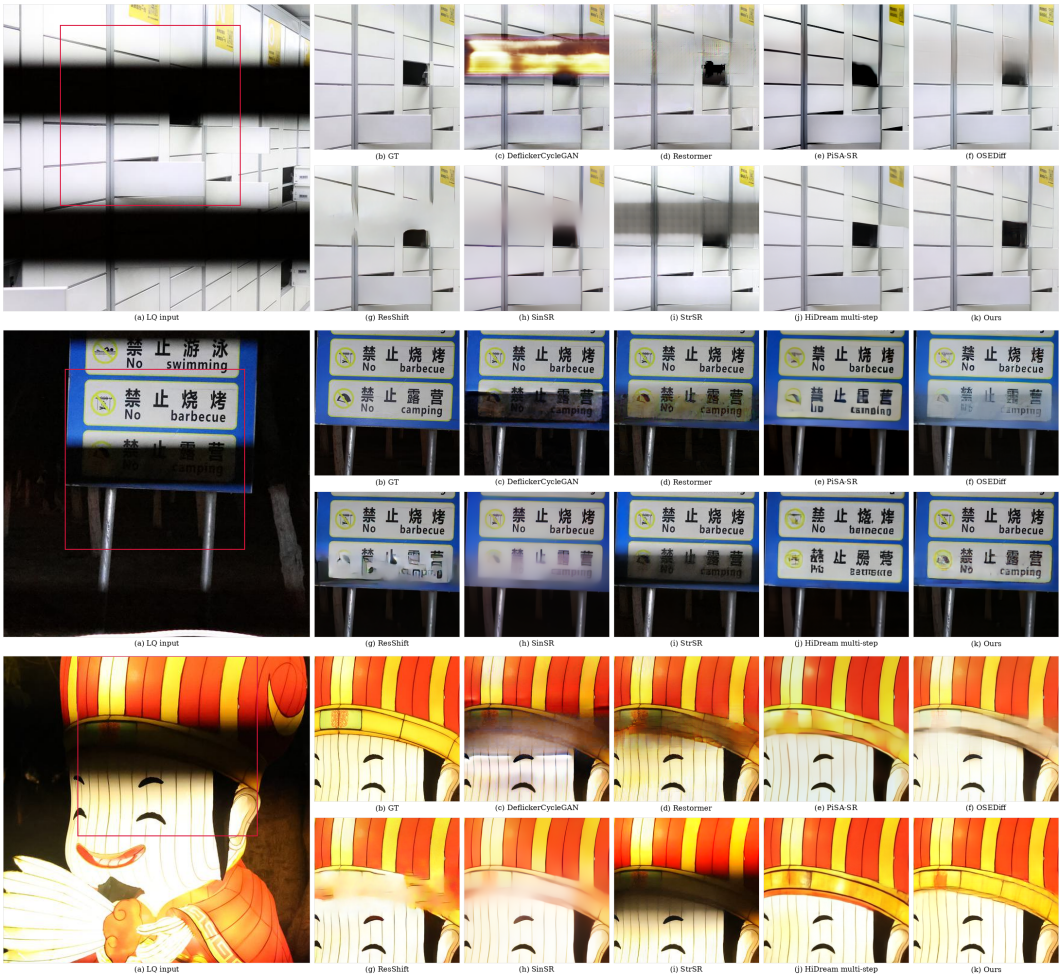


Figure 10: Additional qualitative comparison on BurstFlicker-S. Each example shows a $512 \times 512$ input view spanning two rows on the left, with the crop region marked in red. The right panels show matching square crops from the ground truth, all baselines in Table 1, and FlickerDiff.