LucidNFT: LR-Anchored Multi-Reward Preference Optimization for Flow-Based Real-World Super-Resolution

Song Fei[1†] **Tian Ye**[1†] **Sixiang Chen**[1] **Zhaohu Xing**[1] **Jianyu Lai**[1] **Lei Zhu**[1,2*]
[1]The Hong Kong University of Science and Technology (Guangzhou)
[2]The Hong Kong University of Science and Technology
[†]Equal contribution

Project page: `https://w2genai-lab.github.io/LucidNFT`
Code: `https://github.com/W2GenAI-Lab/LucidNFT`

Abstract

Generative real-world image super-resolution (Real-ISR) can synthesize visually convincing details from severely degraded low-resolution (LR) inputs, yet its stochastic sampling makes a critical failure mode hard to avoid: outputs may look sharp but be *unfaithful* to the LR evidence, exhibiting semantic or structural hallucinations. Once a generative prior has been adapted to Real-ISR, its stochastic sampling turns each LR input into a *rollout group* of candidates with different perceptual–faithfulness trade-offs, which allows reinforcement learning (RL) post-training to optimize objectives that paired HR supervision cannot specify. However, effective alignment in Real-ISR is hindered by three coupled challenges: (i) the lack of an LR-referenced faithfulness signal that is robust to degradation yet sensitive to localized hallucinations, (ii) a rollout-group optimization bottleneck where scalarizing heterogeneous rewards before normalization compresses objective-wise contrasts and weakens the reward-weighted updates of Diffusion Negative-aware FineTuning (DiffusionNFT), and (iii) limited coverage of real degradations, which restricts rollout diversity and preference signal quality. We propose **LucidNFT**, a multi-reward RL framework for *flow-matching* Real-ISR. LucidNFT introduces **LucidConsistency**, a degradation-invariant and hallucination-sensitive LR-referenced evaluator trained with content-consistent degradation pools and original–inpainted hard negatives; a **decoupled reward normalization** strategy that preserves objective-wise contrasts within each LR-conditioned rollout group before fusion; and **LucidLR**, a large-scale collection of real-world degraded images for robust RL fine-tuning. Extensive experiments show that LucidNFT improves perceptual quality on strong flow-based Real-ISR baselines while generally maintaining LR-referenced consistency across diverse real-world scenarios, and a reward-independent human study judges its outputs more LR-faithful in 72.5% of comparisons.

1 Introduction

Real-world image super-resolution (Real-ISR) aims to recover a high-resolution (HR) image from a degraded low-resolution (LR) observation under *unknown* and *heterogeneous* degradations. Empowered by large-scale generative priors, recent generative Real-ISR approaches—including diffusion-based methods [63, 79, 65, 70, 40, 2, 77] and scalable flow-matching formulations [22, 19]—have substantially improved perceptual restoration quality by synthesizing rich and plausible high-frequency details beyond regression-based frameworks [82, 18, 14]. However, this generative freedom also

*Lei Zhu is the corresponding author of this work.

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

exposes a central reliability bottleneck: restored details may look realistic yet be *unfaithful* to the LR evidence (i.e., semantic or structural hallucination [15]). In the absence of HR ground truth, there is no reliable way to *measure*, and therefore to *optimize*, such LR-anchored faithfulness, which is a key obstacle to deploying and aligning generative Real-ISR models in real-world settings.

Stochastic sampling is not only the source of hallucination but also a source of supervision. Across image generation [8], video generation [24], and language modeling [49], a standard recipe first establishes basic capability through supervised training and then applies reinforcement learning (RL) [59] to optimize objectives that direct supervision cannot easily specify. Real-ISR fits this recipe: backbones such as LucidFlux [22] and DiT4SR [19] have already adapted generative priors to restoration, and for a fixed LR input their stochastic sampling yields candidates with different perceptual–faithfulness trade-offs, some adding plausible but LR-unsupported details and others preserving the LR evidence at the cost of visual appeal. Large real LR collections also lack paired HR references, so paired supervision cannot exploit them, whereas rewards on generated candidates can. Preference-based RL post-training is therefore a principled, though not the only, route to promoting candidates that are both perceptually strong and LR-faithful.

Making preference optimization effective for Real-ISR, however, is challenging. Preference learning compares rollouts *conditioned on the same LR input*, and these candidates can excel in different objectives. The common practice of scalarizing multiple rewards into one score before normalization [84, 42, 72] allows the reward with the larger within-group variance to dominate the advantage, so that candidates with distinct perceptual–faithfulness trade-offs receive similar advantages. We build on Diffusion Negative-aware FineTuning (DiffusionNFT) [84], a forward-process RL objective for flow-matching models that is compatible with black-box solvers, is reported to be 3–$25\times$ more efficient than FlowGRPO [42], and has been adopted for image editing, few-step generation, and unified RL toolkits [38, 58, 28, 50]. Like the group-based methods above, DiffusionNFT consumes a single scalar reward, and therefore inherits this *advantage collapse* whenever several rewards are fused, in which case the faithfulness reward can be outweighed by the perceptual one.

Two further conditions are needed for RL alignment to be meaningful. First, no-reference perceptual metrics [30, 62, 68, 60, 80, 73] do not evaluate LR-anchored faithfulness, and optimizing them alone can reward over-sharpening and hallucinated textures; Real-ISR needs an *LR-referenced* faithfulness signal that is robust to unknown degradations. Second, RL relies on informative rollouts, but paired Real-ISR datasets [5, 67] are small and synthetic degradation pipelines [64, 1] do not capture entangled real-world degradations.

To address these challenges, we propose LucidNFT, a multi-reward RL framework for aligning flow-matching Real-ISR models. LucidNFT consists of three components:

- **LucidConsistency**, a degradation-invariant and hallucination-sensitive LR-referenced evaluator, trained with multi-degradation views as positives and original–inpainted pairs as hard negatives.
- **Decoupled reward normalization**, which normalizes each reward within its LR-conditioned rollout group before fusion, so that neither reward dominates the other.
- **LucidLR**, a collection of 20K real-world degraded images that provides diverse LR inputs for informative rollouts, released with per-image license metadata.

LucidNFT does not introduce a new policy objective: it keeps the DiffusionNFT update unchanged and redesigns the step that maps a multi-dimensional reward vector to the scalar reward weight that DiffusionNFT consumes. Experiments on two flow-based Real-ISR backbones show that LucidNFT improves perceptual quality while generally maintaining LR-referenced consistency, and a human study that does not rely on LucidConsistency prefers LucidNFT outputs as more LR-faithful in 72.5% of judgments.

2 Related Work

2.1 Generative Real-World Image Super-Resolution

Recent Real-ISR methods increasingly rely on large-scale generative priors. Diffusion-based approaches [79, 63, 70, 40, 2, 77] leverage pretrained text-to-image models [55, 51, 20, 10, 4] to synthesize perceptually rich details beyond regression-based SR, while flow-based formulations [22, 19] improve scalability and sampling efficiency through velocity-field learning.

2.2 Evaluation Methods for Real-World Image Super-Resolution

In the absence of HR references, full-reference metrics such as PSNR and SSIM [66] are inapplicable to real-world settings, and Real-ISR is commonly evaluated with no-reference image quality assessment [30, 60, 73, 80], including CLIP-based and vision–language evaluators [62, 68]. These metrics measure perceptual quality but not whether details are supported by the LR input (Sec. 4.1).

2.3 Reinforcement Learning for Vision Generation Models

Reinforcement learning has been used to align generative models with external objectives. DDPO [3] optimizes diffusion sampling with PPO-style updates [56], Diffusion-DPO [61] adapts DPO [54] to pairwise diffusion preferences, and DiffusionNFT [84] injects reward signals into the forward flow-matching objective through implicit velocity perturbations, enabling fine-tuning without explicit reverse-trajectory policy-gradient estimation, and has been adopted in several downstream systems [38, 58, 28, 50]. Recent Real-ISR alignment works further explore perceptual preferences, LR-conditioned rewards, and fine-grained reward design [69, 57, 44, 52]. Unlike these works, Lucid-NFT targets the multi-reward setting, in which each LR-conditioned rollout group must be ranked by both perceptual realism and LR faithfulness.

Appendix A discusses further Real-ISR alignment and dataset works.

3 Preliminaries

3.1 Flow Matching

Flow matching [41, 43] learns a continuous transport map between a source distribution and a target data distribution through a velocity field. Given a data sample $x_0 \sim \mathcal{X}_0$ and a Gaussian sample $x_1 \sim \mathcal{X}_1$, rectified flow constructs a linear interpolation $x_t = (1-t)x_0 + tx_1, \quad t \in [0,1]$. The corresponding velocity field is defined as $v = x_1 - x_0$. A neural network $v_\theta(x_t, t, c)$ is trained to approximate this velocity by minimizing the flow-matching objective

$$\mathcal{L}_{\mathrm{FM}}(\theta) = \mathbb{E}_{t,x_0,x_1}\left[\|v - v_\theta(x_t,t,c)\|_2^2\right]. \tag{1}$$

For conditional generation tasks such as Real-ISR, the velocity field is conditioned on an observed input c, learning a transport map from noise to structured outputs guided by the LR observation.

3.2 Diffusion Negative-aware FineTuning

DiffusionNFT [84] performs direct policy optimization on the forward diffusion process by leveraging a scalar optimality reward $r(x_0, c) \in [0,1]$. Unlike standard policy-gradient formulations [42, 72] that optimize reverse-time likelihoods, DiffusionNFT introduces a contrastive diffusion objective that explicitly encourages the velocity predictor to move toward high-reward behaviors and away from low-reward behaviors within the forward flow matching framework. Given the old policy v_{old} that generates the rollouts and is held fixed within an iteration, DiffusionNFT constructs implicit positive and negative policies:

$$v_\theta^+(x_t,t,c) = (1-\beta_{\mathrm{NFT}})v_{\text{old}}(x_t,t,c) + \beta_{\mathrm{NFT}}v_\theta(x_t,t,c), \tag{2}$$

$$v_\theta^-(x_t,t,c) = (1+\beta_{\mathrm{NFT}})v_{\text{old}}(x_t,t,c) - \beta_{\mathrm{NFT}}v_\theta(x_t,t,c), \tag{3}$$

where the interpolation weight β_{NFT} controls the guidance strength; it is distinct from the coefficient λ_{KL} of a separate KL-style regularizer, the squared distance between the velocity predictions of v_θ and of the pretrained reference model. These perturbations can be interpreted as shifting the reference velocity field along the improvement direction $v_\theta - v_{\text{old}}$. The training objective is defined as

$$\mathcal{L}(\theta) = \mathbb{E}_{c,\, x_0 \sim \pi_{\text{old}},\, t}\left[r\|v_\theta^+ - v\|_2^2 + (1-r)\|v_\theta^- - v\|_2^2\right], \tag{4}$$

where v denotes the ground-truth velocity target. High-reward samples emphasize the positive perturbation term, while low-reward samples increase the contribution of the negative term, forming a reward-weighted contrastive update in velocity space. To stabilize optimization, rewards are normalized as

$$r(x_0,c) = \frac{1}{2} + \frac{1}{2}\,\mathrm{clip}\left(\frac{r_{\text{raw}}(x_0,c) - \mathbb{E}_{\pi_{\text{old}}}[r_{\text{raw}}]}{Z_c}, -1, 1\right), \tag{5}$$

where Z_c controls the reward scale and the clipping threshold. In practice, the expectation in Eq. (5) is estimated from a *rollout group*: several samples are drawn for the same condition c, and each reward is compared with the mean of its group, as in group-relative policy optimization [42, 72].

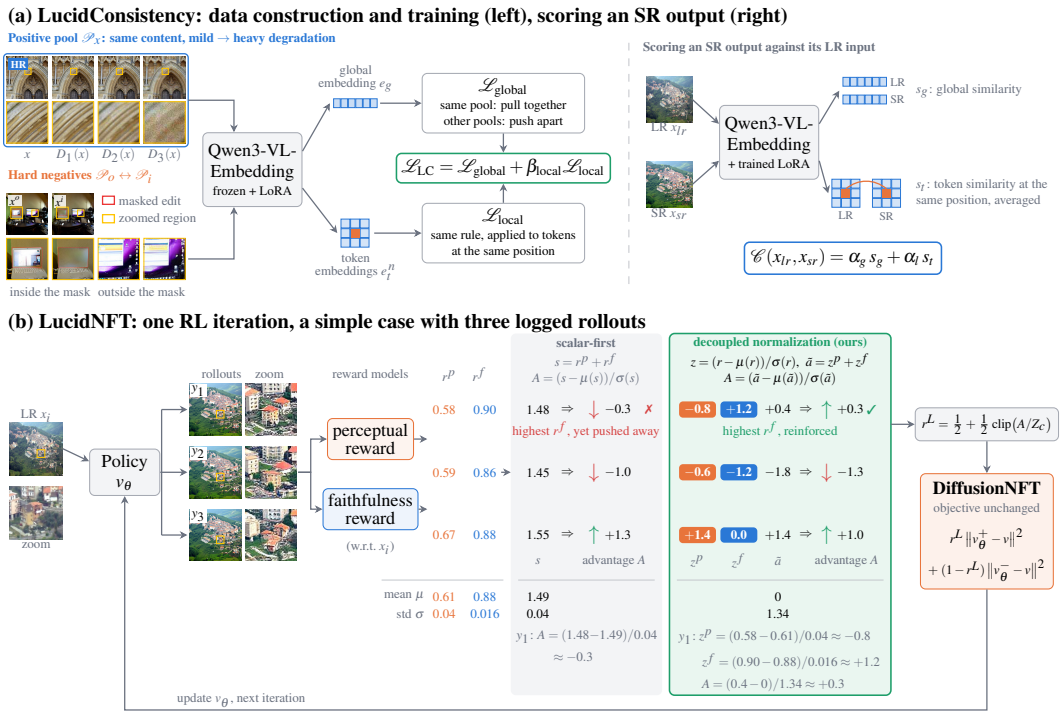

Figure 1: Overview of LucidNFT. (a) LucidConsistency: the positive pool $\mathcal{P}_x$ holds an HR image x (LSDIR) and its degraded views $D_k(x)$ of increasing severity; an original image x^o and its inpainted counterpart x^i (SAGI) form pools $\mathcal{P}_o$, $\mathcal{P}_i$ that are hard negatives for each other: they differ inside the masked region and, more subtly, in local details outside it that the regeneration alters (zooms). The global embedding e_g and token embeddings e_t^n are trained with $\mathcal{L}_{\text{LC}}$ (Eq. (7)); an SR output x_{sr} is then scored against its LR input x_{lr} by $\mathcal{C} = \alpha_g s_g + \alpha_l s_t$, with global similarity s_g and position-averaged token similarity s_t (Eq. (8)). (b) One training iteration of the policy v_θ on three logged rollouts y_j of one LR input x_i (training uses 24 groups of M=12). r^p, r^f: perceptual and faithfulness rewards; s: their sum; μ, σ: mean and standard deviation over the three rollouts; z: a reward standardized within the group; $\tilde{a}$: the sum of the z; A: advantage (Eqs. (9)–(11)), which reinforces an output if positive (↑) and pushes it away if negative (↓); r^L: reward weight with scale Z_c, which weights the implicit positive and negative policies $v_\theta^{\pm}$ against the target velocity v in DiffusionNFT (Eq. (4)). Summing before normalization suppresses the most faithful candidate y_1, whereas decoupled normalization reinforces it.

The resulting advantage is positive for samples that are better than the other samples of the same input, which the update reinforces, and negative for worse ones, which it suppresses; only differences among samples of the same input therefore drive learning. By embedding reward-weighted contrast directly into the forward velocity objective, DiffusionNFT avoids likelihood approximation, supports black-box solvers, and needs only clean samples and their rewards rather than full trajectories. These properties, together with its reported efficiency and downstream adoption (Sec. 1), motivate our choice; in particular, they allow us to retain the backbone's multi-step solver without storing sampling trajectories. DiffusionNFT, however, consumes a *single* bounded reward $r \in [0, 1]$ (Sec. 4.2).

4 Method

LucidNFT is an LR-anchored multi-reward preference optimization framework for flow-matching Real-ISR built on DiffusionNFT (Sec. 3.2): for each LR input from **LucidLR** (Sec. 4.3), the policy samples a rollout group, each candidate is scored by a perceptual IQA reward and by **LucidConsistency** (Sec. 4.1), and **decoupled reward normalization** (Sec. 4.2) maps the reward vectors to the reward weight of the unchanged DiffusionNFT objective (Eq. (4)); Fig. 1 illustrates both stages.

4.1 LucidConsistency: Degradation-Invariant and Hallucination-Sensitive Consistency Evaluation

Existing Real-ISR evaluation protocols mainly rely on no-reference perceptual metrics [30, 62, 68, 60, 80, 73], which assess visual quality but cannot determine whether restored details are supported by the LR observation. A natural alternative is to compare the LR input and the SR output in a generic semantic embedding space, but such embeddings are not trained for this comparison and agree less with human faithfulness judgments (Table 2). An LR-referenced faithfulness signal for Real-ISR must instead satisfy two requirements at once. It must be *degradation-invariant*: an LR input and a faithful SR output differ in resolution, blur, noise, and residual artifacts even when their content agrees, and these differences should not be penalized. It must also be *hallucination-sensitive*:

it should still react to the localized, globally plausible changes that generative SR introduces, such as unsupported text strokes, repeated patterns, or fabricated facial parts. Both requirements are encoded in the training pools: multi-degradation views of an image are positives, and original–inpainted pairs are hard negatives (Fig. 1a). We train LucidConsistency on these pools with standard contrastive learning [48, 31]; the design lies in the supervision that the pools define.

Degradation invariance: content-consistent pools. For an image x, we build a pool $\mathcal{P}_x = \{x, D_1(x), \ldots, D_K(x)\}$ of degraded views produced by a randomized Real-ESRGAN pipeline at increasing severity, from mild to heavy; within each level, blur kernel, resizing factor, noise strength, and JPEG quality are sampled at random (ranges in Appendix B.1). Since all views share content and geometry, treating them as positives trains the evaluator to ignore degradation-induced shifts.

Hallucination sensitivity: paired hard negatives. Invariance to degradation alone would leave the evaluator insensitive to hallucinated content; the training data must therefore also contain negatives that differ from their references only in localized, generator-induced content. Outputs of SR models are ill-suited to this role: their unfaithful regions are not annotated, and training on them would couple the evaluator to the biases of the restoration models that produced them. We therefore draw paired hard negatives from SAGI [25], a large-scale dataset of AI-inpainted images. Each pair consists of an original image x^o and its inpainted counterpart x^i; the two are spatially aligned and globally similar but differ within a regenerated region, which mirrors the failure mode of generative SR. We use the fully regenerated subset of SAGI, in which the inpainted images are produced by BrushNet [29], ControlNet inpainting [81], HD-Painter [47], and PowerPaint [85] on checkpoints of the SD 1.5, SD 2, and SDXL families, as well as by two successive edits that may additionally involve Inpaint Anything [78]; the hallucinated content thus originates from a diverse set of generators (statistics in Sec. 5.1). Both images are expanded into degraded pools, $\mathcal{P}_o = \{x^o, D_1(x^o), \ldots, D_K(x^o)\}$ and $\mathcal{P}_i = \{x^i, D_1(x^i), \ldots, D_K(x^i)\}$, which serve as hard negatives for each other. As the degradation pipeline and the inpainting generators are unrelated, invariance and sensitivity are learned from independent sources of variation; Sec. 5.2 examines robustness to the choice of inpainting generators with a leave-one-inpainter-out protocol.

Global-local representation. We use Qwen3-VL-Embedding [35] as the backbone and train only LoRA adapters. For each image view v, we extract a global embedding $e_g(v)$ and native image-token embeddings $E_t(v) = \{e_t^i(v)\}_{i=1}^{N_v}$, where N_v is the number of image tokens produced for v. The global embedding captures image-level semantic consistency, while native tokens provide spatially indexed local correspondence without manually defining crops or windows. All views are resized to a shared spatial size before encoding, so $N_v = N$ within a batch and same-index tokens correspond to the same backbone-defined visual positions.

Pool-based contrastive learning. For each query representation q, positives are representations from other views in the same pool, while negatives are representations from other pools. Let $\mathcal{Q}$ be the set of valid queries in a mini-batch, $\mathcal{P}^+(q)$ the same-pool positive set excluding q, and $\mathcal{P}^-(q)$ the negative set from other pools. We use

$$\mathcal{L}_{\text{ctr}} = -\frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \log \frac{\sum\limits_{p \in \mathcal{P}^+(q)} \exp\left(\mathrm{s}(q,p)/\tau\right)}{\sum\limits_{p \in \mathcal{P}^+(q)} \exp\left(\mathrm{s}(q,p)/\tau\right) + \sum\limits_{n \in \mathcal{P}^-(q)} \exp\left(\mathrm{s}(q,n)/\tau\right)}. \tag{6}$$

For the global branch, q, p, n are global embeddings. For the local branch, q is a token embedding $e_t^i(\cdot)$, positives are same-index tokens from same-pool views, and negatives are same-index tokens from other pools. Applying Eq. (6) to the two branches gives

$$\mathcal{L}_{LC} = \mathcal{L}_{global} + \beta_{local}\mathcal{L}_{local}. \tag{7}$$

Inference score. Given an LR input x_{lr} and a generated SR output x_{sr}, LucidConsistency combines global and local consistency:

$$\mathcal{C}(x_{lr}, x_{sr}) = \alpha_g \, \mathrm{s}\big(e_g(x_{lr}), e_g(x_{sr})\big) + \alpha_l \, \frac{1}{N} \sum_{i=1}^{N} \mathrm{s}\big(e_t^i(x_{lr}), e_t^i(x_{sr})\big), \qquad \alpha_g + \alpha_l = 1. \tag{8}$$

In all experiments, we set $\tau = 0.7$, $\beta_{\text{local}} = 1.0$, and use equal global-local weights $\alpha_g = \alpha_l = 0.5$ unless otherwise specified. This score serves as both an evaluation metric and an RL reward.

4.2 Decoupled Reward Normalization for Rollout-Group Real-ISR

For a fixed LR input, stochastic sampling produces a rollout group of candidate restorations that differ along the perceptual–faithfulness trade-off: a sharper output may hallucinate unsupported details, while a more conservative one may better preserve the LR evidence. The useful supervision is therefore the reward contrast among candidates of the same LR input, rather than the absolute reward of each sample. DiffusionNFT, however, consumes this contrast through a single bounded reward weight $r \in [0, 1]$ (Sec. 3.2), whereas Real-ISR must balance at least two heterogeneous rewards. The bottleneck we address lies at this scalar-reward interface: how a multi-dimensional reward vector is mapped to r decides whether both contrasts survive.

Notation. Let x_i denote the i-th LR input and $y_{i,j}$ its j-th rollout within the same LR-conditioned group. For each rollout, we compute a K-dimensional reward vector $\mathbf{r}_{i,j} = (r_{i,j}^{(1)}, \ldots, r_{i,j}^{(K)})$, where each dimension corresponds to one objective and has weight λ_k. We write μ_i, σ_i for the mean and standard deviation within the group of x_i, and μ_B, σ_B for those over the training batch.

Scalar-first aggregation. A common strategy first fuses rewards into a scalar score $s_{i,j} = \sum_{k=1}^{K} \lambda_k r_{i,j}^{(k)}$, and then normalizes the scalar scores within rollout groups:

$$A_{i,j} = \frac{s_{i,j} - \mu_i(s)}{\sigma_B(s) + \epsilon}, \tag{9}$$

where $\mu_i(s) = \frac{1}{M_i} \sum_{j=1}^{M_i} s_{i,j}$.

Failure mode of scalar-first fusion. Consider a group of three candidates with (perceptual, faithfulness) rewards $(1, 0)$, $(0.5, 0.5)$, and $(0, 1)$. With $\lambda_1 = \lambda_2 = 1$, all three obtain $s_{i,j} = 1$ and therefore $A_{i,j} = 0$: the group contains two opposite orderings, yet the update receives no preference signal. Normalizing each reward first yields $z^{(1)} = (+1.22, 0, -1.22)$ and $z^{(2)} = (-1.22, 0, +1.22)$, which preserves both orderings; with equal weights this symmetric case still sums to zero. Real rollout groups are rarely this symmetric; in practice, scalar-first fusion instead reweights the objectives implicitly. On 16 held-out LR inputs with 12 LucidFlux rollouts each, both rewards lie in $[0, 1]$ and carry nominal weights 1:1, yet their within-group standard deviations differ by a factor of $0.0305/0.0104 \approx 2.94$ (Fig. 2a). Under scalar-first fusion, the fused advantage correlates with the perceptual reward at 0.861 but with the faithfulness reward at only 0.318, so the effective weighting is closer to 2.7:1 than to the specified 1:1 (Fig. 2b). Fig. 1b applies both fusion rules to the logged rewards of three rollouts of one training input: scalar-first fusion would suppress the most faithful candidate, whereas decoupled normalization reinforces it. As DiffusionNFT maps advantages to a bounded weight $r \in [0, 1]$, the under-weighted objective also occupies little of the range around $r = 0.5$. We refer to this as *advantage compression*.

Decoupled normalization before fusion. LucidNFT preserves objective-wise contrasts by normalizing each reward dimension within the same LR-conditioned rollout group before fusion:

$$z_{i,j}^{(k)} = \frac{r_{i,j}^{(k)} - \mu_i(r^{(k)})}{\sigma_i(r^{(k)}) + \epsilon}, \tag{10}$$

We then fuse the normalized rewards as $\tilde{a}_{i,j} = \sum_{k=1}^{K} \lambda_k z_{i,j}^{(k)}$. Finally, we standardize $\tilde{a}$ over the batch and map the result to the bounded DiffusionNFT reward weight:

$$A_{i,j}^{L} = \frac{\tilde{a}_{i,j} - \mu_B(\tilde{a})}{\sigma_B(\tilde{a}) + \epsilon}, \qquad r_{i,j}^{L} = \frac{1}{2} + \frac{1}{2} \operatorname{clip}\left(\frac{A_{i,j}^{L}}{Z_c}, -1, 1 \right). \tag{11}$$

Since each $z^{(k)}$ is standardized within its group, $\mu_B(\tilde{a}) = 0$ and this step only rescales $\tilde{a}$. We substitute $r_{i,j}^{L}$ for the reward weight in the DiffusionNFT objective and keep all other terms unchanged. On the same rollouts, the correlations of the fused advantage with the perceptual and faithfulness rewards become 0.680 and 0.674, so the two objectives contribute almost equally, as the 1:1 weights specify (Fig. 2b).

Resolution of the training signal. Preserving both orderings also makes the advantages more informative. Fig. 2c counts the distinct advantage levels within a group on the same rollouts: with 12 rollouts, decoupled normalization separates the candidates into 8.0 levels on average, whereas scalar-first fusion yields 3.8, leaving many candidates with different trade-offs at the same advantage. The advantage gap between the two most different candidates grows accordingly, from 0.72 to 2.19 at M=12 (Appendix C.2).

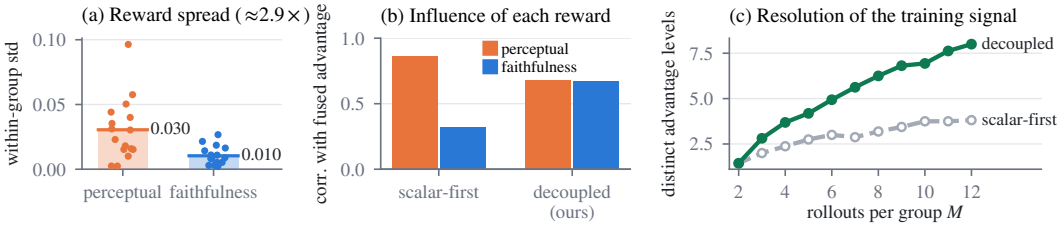

Figure 2: Failure mode of scalar-first fusion (16 LR inputs × 12 LucidFlux rollouts). (a) The perceptual reward varies about 3× more within a group than the faithfulness reward. (b) Summing before normalization therefore yields advantages that follow the perceptual reward, whereas decoupled normalization gives both rewards equal influence (within-group Pearson correlation). (c) It also separates more candidates.

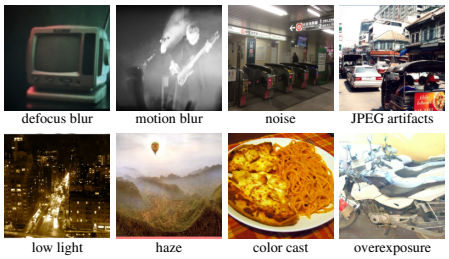

Figure 3: LucidLR inputs for Real-ISR: real low-quality photographs, each labeled with the dominant degradation that accompanies its low resolution.

Table 1: Real-world Real-ISR datasets. Degradation statistics use the same annotator and taxonomy for all datasets (Appendix E.2): number of primary-degradation types observed (excluding *other* and *none*), and share of images whose primary degradation is defocus blur.

Dataset	Pairing	Usage	# Images	Degr. types	Defocus (%)
RealSR [5]	Paired	Benchmark	100	4	82.0
DRealSR [67]	Paired	Benchmark	93	5	91.4
RealLQ250 [2]	Unpaired	Benchmark	250	10	43.2
LucidLR (ours)	Unpaired	RL training	20,000	15	26.5

4.3 LucidLR: A Large-Scale Real-World Degradation Dataset for Real-ISR

Existing real-world Real-ISR datasets [5, 67, 2] are small benchmarks (Table 1), whereas RL fine-tuning benefits from a large and diverse set of real degradations that induce informative rollout variation. We therefore construct **LucidLR** from the Wikimedia Commons categories *Images of low quality* and *Blurred images*, whose photographs exhibit naturally occurring degradations and may be reused with attribution. The raw pool contains 21.7K unique images. A two-pass screening with a vision-language model (DeepSeek-V4.1-Flash [17]) removes 72 images containing explicit sexual content, graphic violence, personal data, or hate symbols; we further exclude images whose short side is below 128 pixels or whose source pages are no longer available, and supplement the pool with images collected under the same protocol, yielding **20,000** images (Fig. 3). LucidLR contains only images whose licenses permit redistribution. It is released both as an image bundle and as a metadata index with a reference downloader, and each image is accompanied by a JSON sidecar that records its source, author, and license (Appendix E.1). Annotated with the same model and taxonomy as existing benchmarks, LucidLR spans all 13 semantic categories and 15 primary-degradation types, and far fewer of its images are dominated by defocus blur (Table 1); its images are not resized and vary widely in resolution (median 1080×960; Appendix E.2).

5 Experiments

5.1 Implementation Details

For **LucidConsistency**, we adopt Qwen3-VL-Embedding-8B [35] as the backbone and train LoRA adapters while keeping the backbone frozen. The training data consist of two parts. First, 85K LSDIR [37] images are used to build content-consistent pools for degradation-invariant alignment, where multi-level views are generated by the Real-ESRGAN [64] degradation pipeline. We use three ordered degradation levels (mild, medium, and heavy), with detailed parameter ranges provided in Appendix B.1. Second, original–inpainted hard negatives come from SAGI [25]: we use all fully regenerated pairs of its training split whose originals are from COCO [39] and RAISE [16], 37.7K pairs from 33.3K originals. Of these, 27.8K pairs are edited once, by one of 11 method–checkpoint combinations that SAGI provides, i.e., each inpainting method on the checkpoints it supports (Fig. 1a): BrushNet [29] on SD 1.5 [55], DreamShaper, Realistic Vision, epiCRealism, SDXL [51], and Juggernaut XL; ControlNet inpainting [81] on SD 1.5; HD-Painter [47] on SD 1.5, SD 2, and DreamShaper; and PowerPaint [85] on Realistic Vision. These seven checkpoints belong to three base families: SD 1.5 and its fine-tunes, SD 2, and SDXL with its fine-tune Juggernaut XL; the remaining 9.9K pairs are edited twice in succession, each round by one of these four methods or by Inpaint Anything [78] on SD 2 (2.9K pairs). Every inpainted variant forms its own pool, a hard negative for its original and an ordinary negative for all other pools; each pair is brought to a common pixel grid and cropped around the edit (Appendix B.1). We train LucidConsistency at resolution

Table 2: Human-aligned LR-faithfulness evaluation on RealSR. Expert annotators rank SR candidates by faithfulness to the LR input rather than perceptual quality.

Evaluator	Criterion	Agreement	Recall@1	Filter@1
CLIP-IQA	Perceptual Quality	0.391	0.186	0.093
MUSIQ	Perceptual Quality	0.349	0.116	0.093
Q-Align	Perceptual Quality	0.322	0.093	0.047
UniPercept-IQA	Perceptual Quality	0.322	0.070	0.093
Qwen3-VL-Embedding-8B	Generic Semantics	0.643	0.465	0.302
LucidConsistency	LR Faithfulness	**0.690**	**0.558**	**0.558**

Table 3: Controlled ablation on RealLQ250 (LucidFlux). First row: absolute values; other rows: changes relative to it (larger is better). Each row modifies only the stated variable.

Method	UniPercept ↑	VQ-R1 ↑	NIQE ↓	LucidConsistency ↑
LucidFlux (backbone)	70.930	4.547	3.741	0.9237
DiffusionNFT baselines				
(A) UniPercept IQA reward	+0.608	+0.054	−0.227	+0.0022
(B) (A) + frozen Qwen3-VL reward	+0.436	+0.049	−0.212	+0.0061
Our components				
(C) (B), semantic → LucidConsistency	+0.284	+0.046	−0.194	+0.0104
(D) (C), LSDIR → LucidLR	+1.40	+0.062	−0.345	+0.0114
(E) (C), scalar-first → decoupled	+1.773	+0.086	−0.369	**+0.0119**
(F) (E), LSDIR → LucidLR (Full)	**+2.549**	**+0.104**	**−0.488**	+0.0108

1024 with LoRA rank and alpha both 128, using batch size 2 on 2 NVIDIA A100 GPUs for 6,000 steps. The learning rate is 1×10^{-4}. For **LucidNFT**, we fine-tune the base Real-ISR model using LoRA [26] (rank/alpha 32/64) on the LucidLR dataset. Training runs on 8 NVIDIA A100 GPUs for 600 RL steps with a learning rate of 3×10^{-5}, using 24 LR groups of 12 rollouts per step at resolution 768; solver, β_{NFT}, λ_{KL}, EMA, and the remaining hyperparameters are listed with pseudocode in Appendix C.1. We use UniPercept IQA [7] as the perceptual reward and LucidConsistency as the LR-faithfulness reward, both with weight 1.0. Both models are optimized with AdamW [45].

5.2 Human-Aligned Faithfulness Evaluation

We further evaluate whether LucidConsistency reflects human judgments of LR-conditioned faithfulness. On RealSR [5], we collect SR candidates from eight generative SR methods: ResShift [79], SeeSR [70], LucidFlux [22], DreamClear [2], SUPIR [77], DiffBIRv2 [40], DiT4SR [19], and StableSR [63]. For each LR input, five expert annotators rank four randomly sampled candidates by faithfulness to the LR evidence rather than sharpness or aesthetics, yielding six pairwise comparisons (instructions and interface in Appendix B.2).

We compare evaluator-induced rankings with human preferences using pairwise Agreement, Recall@1 (selecting the human-preferred best candidate), and Filter@1 (assigning the lowest score to the human-labeled worst candidate). As shown in Table 2, LucidConsistency achieves the best results on all three metrics. Compared with the frozen semantic baseline, it improves Agreement from 0.643 to 0.690, Recall@1 from 0.465 to 0.558, and Filter@1 from 0.302 to 0.558, indicating that pool-based training better calibrates generic semantic representations toward LR-conditioned faithfulness. Perceptual metrics perform substantially worse under this protocol, suggesting that visual quality alone is insufficient for identifying hallucinated or LR-inconsistent details.

Robustness to the hard-negative construction. To test whether LucidConsistency overfits to its synthetic inpainting negatives, we retrain it without every pair in which an SDXL-family generator (BrushNet with SDXL or Juggernaut XL) takes part in either editing round (leave-one-inpainter-out, LOIO), thereby withholding an entire base architecture. On the SAGI test split of the held-out generators and on the 40 human-labeled RealSR pairs of Sec. 5.3, the paired agreement drops only from 0.74 and 0.69 (full) to 0.70 and 0.67 (LOIO), both well above chance, which suggests that the learned faithfulness signal does not hinge on the generators seen in training (Appendix B.3).

5.3 Effectiveness of LucidNFT

We evaluate LucidNFT on two representative flow-based Real-ISR models, LucidFlux [22] and DiT4SR [19]. LucidFlux is a 14B DiT-based model with a 12B flow prior, whereas DiT4SR is a lighter DiT derived from Stable-Diffusion-3-Medium [20] (2B prior and 2.72B adapter), allowing us to test LucidNFT across backbone scales.

During training, the validation UniPercept IQA reward of LucidNFT rises steadily on both backbones while LucidConsistency stays stable (Appendix Fig. 7).

Comparison Methods and Evaluation Metrics. We compare LucidFlux(+LucidNFT) with representative generative SR methods, including DiffBIRv2 [40], SeeSR [70], DreamClear [2], SUPIR [77], DiT4SR [19], and LucidFlux [22]. We also include DiT4SR(+LucidNFT) and LucidFlux(+DPO), implemented with Diffusion-DPO [61], to compare against direct preference optimization under related backbones. We use this same-backbone DPO implementation as a controlled direct-preference baseline; DP^{2}O-SR [69] is closely related but its public release does not include training code, as discussed in Appendix A. All methods are evaluated at 1024×1024 output resolution with $4\times$ upscaling, following official inference settings when available. We report eight no-reference quality metrics, including CLIP-IQA+ [62], Q-Align [68], MUSIQ [30], MANIQA [73], CLIP-IQA [62], NIQE [80], UniPercept IQA [7], and VisualQuality-R1 (VQ-R1) [71]. We additionally report LucidConsistency as an LR-referenced consistency score without HR ground truth.

Table 4: Quantitative comparison with state-of-the-art Real-ISR methods on RealLQ250 [2], DRealSR [67], and RealSR [5]. Higher is better for all metrics except NIQE. Values in parentheses are changes over the corresponding backbone. Best and second-best scores are highlighted in bold and underline, respectively. For the parenthetical changes, † and ‡ mark the largest and second-largest improvements among the RL variants; for NIQE, larger improvement means a larger reduction.

Benchmark	Metric	DiffBIRv2	SeeSR	DreamClear	SUPIR	DiT4SR	DiT4SR(+LucidNFT)	LucidFlux	LucidFlux(+DPO)	LucidFlux(+LucidNFT)
RealLQ250	CLIP-IQA+ ↑	0.6919	0.7034	0.6813	0.6532	0.7098	0.7124 (+0.0026)‡	0.7208	0.7228 (+0.0020)	**0.7465** (+0.0257)†
	Q-Align ↑	3.9755	4.1423	4.0647	4.1347	4.2270	4.2358 (+0.0088)	4.4052	4.4430 (+0.0378)‡	**4.4855** (+0.0803)†
	MUSIQ ↑	67.5313	70.3757	67.0899	65.8133	71.6682	72.1732 (+0.5050)‡	72.3351	72.4504 (+0.1153)	**73.4475** (+1.1124)†
	MANIQA ↑	0.4900	0.4895	0.4405	0.3826	0.4607	0.4719 (+0.0112)‡	0.5227	0.5258 (+0.0031)	**0.5443** (+0.0216)†
	CLIP-IQA ↑	0.7137	0.7063	0.6957	0.5767	0.7141	**0.7355** (+0.0214)‡	0.6855	0.6917 (+0.0062)	0.7233 (+0.0378)†
	NIQE ↓	5.1193	4.4383	3.8709	3.6591	3.5556	3.5007 (-0.0549)‡	3.7410	3.7785 (+0.0375)	**3.2532** (-0.4878)†
	UniPercept IQA ↑	65.4760	69.2015	68.8465	68.6430	73.0740	73.3430 (+0.2690)‡	70.9300	71.1330 (+0.2030)	**73.4790** (+2.5490)†
	VisualQuality-R1 ↑	4.3428	4.5118	4.4430	4.4265	4.6146	4.6304 (+0.0158)	4.5474	4.5644 (+0.0170)‡	**4.6510** (+0.1036)†
	LucidConsistency ↑	**0.9609**	0.9466	0.9578	0.9522	0.9052	0.9172 (+0.0120)†	0.9237	0.9296 (+0.0059)	0.9345 (+0.0108)‡
DRealSR	CLIP-IQA+ ↑	0.6476	0.6258	0.4462	0.5494	0.6537	0.6757 (+0.0220)‡	0.6516	0.6530 (+0.0014)	**0.6867** (+0.0351)†
	Q-Align ↑	3.0487	3.2746	2.4214	3.4722	3.6008	3.6641 (+0.0633)‡	3.7141	3.7408 (+0.0267)	**3.8423** (+0.1282)†
	MUSIQ ↑	60.0759	61.3222	35.1912	54.9280	63.8051	65.1915 (+1.3864)‡	64.6025	64.5607 (-0.0418)	**68.1545** (+3.5520)†
	MANIQA ↑	0.4900	0.4505	0.2676	0.3483	0.4419	0.4572 (+0.0153)‡	0.4678	0.4669 (-0.0009)	**0.5004** (+0.0326)†
	CLIP-IQA ↑	0.6782	0.6760	0.4361	0.5310	0.6732	**0.7111** (+0.0379)‡	0.6673	0.6713 (+0.0040)	0.7073 (+0.0400)†
	NIQE ↓	6.4853	6.4503	7.0164	5.9092	5.7001	5.6329 (-0.0672)‡	5.0742	5.0143 (-0.0599)	**4.1788** (-0.8954)†
	UniPercept IQA ↑	46.2298	50.3414	34.2473	55.1371	58.1290	59.9328 (+1.8038)‡	59.9032	59.7782 (-0.1250)	**63.7782** (+3.8750)†
	VisualQuality-R1 ↑	3.4796	3.6116	2.5655	3.7349	3.9603	4.0239 (+0.0636)‡	3.9955	3.9828 (-0.0127)	**4.1455** (+0.1500)†
	LucidConsistency ↑	0.9332	0.9275	**0.9607**	0.8911	0.8438	0.8544 (+0.0106)†	0.8813	0.8890 (+0.0077)‡	0.8879 (+0.0066)
RealSR	CLIP-IQA+ ↑	0.6543	0.6731	0.5331	0.5640	0.6753	0.6881 (+0.0128)‡	0.6669	0.6695 (+0.0026)	**0.7151** (+0.0482)†
	Q-Align ↑	3.3156	3.6073	3.0040	3.4682	3.7106	3.7959 (+0.0853)‡	3.8728	3.9147 (+0.0419)	**3.9918** (+0.1190)†
	MUSIQ ↑	61.7751	67.5660	49.4766	55.6807	67.9828	69.1092 (+1.1264)‡	67.8962	67.9362 (+0.0400)	**70.5625** (+2.6663)†
	MANIQA ↑	0.4745	0.5087	0.3092	0.3426	0.4533	0.4654 (+0.0121)‡	0.4889	0.4907 (+0.0018)	**0.5284** (+0.0395)†
	CLIP-IQA ↑	0.6806	**0.6993**	0.5390	0.4857	0.6631	0.6963 (+0.0332)‡	0.6359	0.6427 (+0.0068)	0.6936 (+0.0577)†
	NIQE ↓	6.0700	5.4594	5.2873	5.2819	5.0912	4.8332 (-0.2580)‡	4.8134	4.6804 (-0.1330)	**3.9526** (-0.8608)†
	UniPercept IQA ↑	53.6550	58.0538	46.7850	56.6063	63.2025	**64.8425** (+1.6400)‡	60.0925	60.4775 (+0.3850)	64.7588 (+4.6663)†
	VisualQuality-R1 ↑	3.8928	4.0635	3.5028	3.7821	4.1953	4.2429 (+0.0476)‡	4.1376	4.1503 (+0.0127)	**4.3389** (+0.2013)†
	LucidConsistency ↑	**0.9544**	0.9138	0.9475	0.9141	0.8318	0.8498 (+0.0180)†	0.8853	0.8932 (+0.0079)‡	0.8923 (+0.0070)

Quantitative Comparison. Table 4 reports quantitative results on RealLQ250 [2], DRealSR [67], and RealSR [5]. Both DiT4SR(+LucidNFT) and LucidFlux(+LucidNFT) improve their corresponding backbones on most perceptual metrics while generally maintaining or improving LucidConsistency. On LucidFlux, LucidNFT improves all eight no-reference metrics on all three benchmarks, with the largest relative gains on NIQE (reductions of 13.0%, 17.7%, and 17.9% on RealLQ250, DRealSR, and RealSR) and MANIQA. LucidFlux(+DPO), which shares the backbone and training data, provides a comparison with direct preference optimization: averaged over the three benchmarks, LucidNFT improves the eight no-reference metrics by 2.8% (Q-Align) to 16.2% (NIQE reduction) and LucidConsistency by 0.9%, whereas Diffusion-DPO changes the perceptual metrics by no more than about 1% on average and degrades 5 of the 24 benchmark–metric entries, all of which LucidNFT improves (Appendix D.2).

Model-level human evaluation. As LucidConsistency also serves as a training reward, its gains in Table 4 are not fully external evidence of faithfulness. We therefore run a human study without LucidConsistency scores: on 40 held-out RealSR inputs, five expert annotators see the LR image and the outputs of LucidFlux and LucidFlux(+LucidNFT) in random order, and select the output better supported by the LR content, disregarding sharpness, saturation, and aesthetics. Over 200 binary judgments, LucidFlux(+LucidNFT) is preferred in 72.5% (Fleiss' $\kappa = 0.415$, moderate agreement [34]; Appendix D.1).

Qualitative Comparison. As shown in Fig. 4, LucidNFT recovers more accurate text strokes and finer local textures than its backbones without visibly adding LR-inconsistent content; more examples are given in Appendix D.4.

5.4 Ablation Study

Table 3 is a controlled ablation on RealLQ250 with LucidFlux, using the same 600-step budget and evaluation as Table 4. By default, LR inputs come from LSDIR [37] images degraded by Real-ESRGAN [64] and rewards are fused scalar-first; (A) and (B) are DiffusionNFT baselines with off-the-shelf rewards, and each later row changes a single variable. *LucidConsistency.* (B)→(C) raises the LucidConsistency gain from $+0.0061$ to $+0.0104$ at a small perceptual cost; since this metric also serves as the reward, the external evidence that it tracks faithfulness better than the frozen embedding is the human alignment in Table 2. *Decoupled normalization.* (C)→(E) lifts the UniPercept gain from $+0.284$ to $+1.773$ and the NIQE reduction from 0.194 to 0.369 while consistency also rises, suggesting that the trade-off in (A)–(C) stems largely from scalar-first fusion rather than from the rewards themselves; the training curves of (A), (C), and (E) show the same ordering throughout optimization (Appendix C.3). *LucidLR.* (C)→(D) changes only the training pool and improves all four metrics, indicating that LucidLR is beneficial even without decoupled normalization; combining both in (F) yields the best perceptual results.

Robustness to reward weights and reward sets. Table 5 compares both fusion schemes under five reward setups with a 300-step budget: three weightings of UniPercept IQA and LucidConsistency,

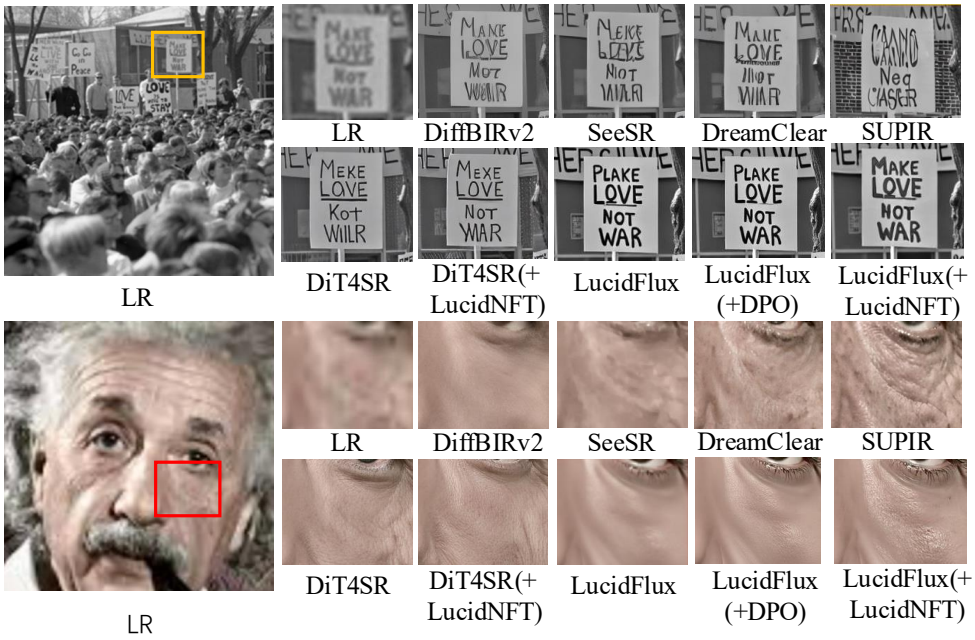

Figure 4: Visual comparison on RealLQ250 [2]; LucidNFT variants recover more accurate text and finer textures than the baselines.

Table 5: Scalar-first (S) versus decoupled (D) fusion under five reward setups (300 steps, RealLQ250). Entries are gains over LucidFlux; the larger gain of each pair is in bold. "Wins" counts the six metrics (Appendix D.3) on which decoupled fusion gains more.

Reward setup	**UniPercept IQA** ↑		**CLIP-IQA** ↑		**MUSIQ** ↑		**MANIQA** ↑		**LucidConsistency** ↑		**Wins**
	S	D	S	D	S	D	S	D	S	D	
UP-IQA : LC = 1:1	+1.21	**+1.69**	+0.016	**+0.028**	+0.02	**+0.55**	−0.002	**+0.011**	+0.0144	**+0.0170**	6/6
UP-IQA : LC = 2:1	+1.01	**+1.70**	+0.009	**+0.027**	−0.03	**+0.69**	−0.006	**+0.013**	**+0.0183**	+0.0157	5/6
UP-IQA : LC = 1:2	+0.78	**+0.92**	+0.009	**+0.018**	−0.34	**+0.10**	−0.010	**+0.002**	**+0.0175**	+0.0169	5/6
+ CLIP-IQA (1:1:1)	+0.46	**+1.31**	+0.004	**+0.027**	−0.43	**+0.39**	−0.013	**+0.011**	+0.0163	**+0.0176**	6/6
+ Qwen3-VL-Sim (1:1:1)	+0.91	**+0.95**	+0.010	**+0.016**	−0.48	**+0.15**	−0.010	**+0.005**	+0.0163	**+0.0172**	6/6

and two extensions with an additional perceptual (CLIP-IQA) or semantic (Qwen3-VL-Sim) reward. Decoupled normalization yields the larger gain on at least five of six metrics in every setup; scalar-first fusion is ahead only on LucidConsistency under unbalanced weights, where some of its perceptual gains (MUSIQ, MANIQA) turn negative (full results in Appendix D.3).

6 Conclusion

We presented LucidNFT, which aligns flow-based Real-ISR models toward both perceptual realism and LR faithfulness through an LR-referenced evaluator, decoupled reward normalization, and a large collection of real degraded inputs; it improves perceptual quality on two backbones while a reward-independent human study judges its outputs more LR-faithful.

Discussion. Our results suggest that hallucination in Real-ISR is capability-dependent: conservative models rarely alter content, whereas stronger models trade LR faithfulness for realism, so hallucination grows more critical as priors improve.

Limitations. (i) No reward can compensate for missing evidence under extreme degradations (severe underexposure, heavy blur, or missing regions). (ii) Both rewards are learned models that prolonged RL may exploit; the bounded reward weight, KL regularization, and a fixed budget mitigate but do not eliminate reward hacking. (iii) LucidLR is biased toward web photography and underrepresents domains such as medical, satellite, or industrial imagery. (iv) LucidConsistency depends on the frozen Qwen3-VL-Embedding-8B backbone; another backbone requires retraining its LoRA adapters, without a guarantee of comparable performance. (v) LucidConsistency is both reward and metric, and no standard reference-free LR-faithfulness benchmark exists; our model-level human study only partially provides external evidence.

Acknowledgments and Disclosure of Funding

This work is supported by Guangdong Basic and Applied Basic Research Foundation (2025A1515140021), and Program of Guangdong Education Department, and Guangdong Provincial Key Lab of Integrated Communication, Sensing and Computation for Ubiquitous Internet of Things(No.2023B1212010007).

References

[1] Agustsson, E., Timofte, R.: NTIRE 2017 challenge on single image super-resolution: Dataset and study. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 126–135 (2017)

[2] Ai, Y., Zhou, X., Huang, H., Han, X., Chen, Z., You, Q., Yang, H.: DreamClear: High-capacity real-world image restoration with privacy-safe dataset curation. Advances in Neural Information Processing Systems **37**, 55443–55469 (2024)

[3] Black, K., Janner, M., Du, Y., Kostrikov, I., Levine, S.: Training diffusion models with reinforcement learning. In: International Conference on Learning Representations (2024)

[4] Black Forest Labs: FLUX (2024)

[5] Cai, J., Zeng, H., Yong, H., Cao, Z., Zhang, L.: Toward real-world single image super-resolution: A new benchmark and a new model. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3086–3095 (2019)

[6] Cai, M., Li, S., Li, W., Huang, X., Chen, H., Hu, J., Wang, Y.: DSPO: Direct semantic preference optimization for real-world image super-resolution. arXiv preprint arXiv:2504.15176 (2025)

[7] Cao, S., Li, J., Li, X., Pu, Y., Zhu, K., Gao, Y., Luo, S., Xin, Y., Qin, Q., Zhou, Y., Chen, X., Zhang, W., Fu, B., Qiao, Y., Liu, Y.: UniPercept: Towards unified perceptual-level image understanding across aesthetics, quality, structure, and texture. In: International Conference on Machine Learning (2026)

[8] Chen, D., Wei, F., Wan, Z., Chen, D., Zhang, J., Zhao, J., Zhang, S., Yue, Y., Liang, Z., Guo, B., Luo, C., Bao, J., Li, J., Shi, L., Yang, Q., Wu, X., Feng, X., Lu, Y., Dong, Y., Wang, Y., Chen, Y.: Lens: Rethinking training efficiency for foundational text-to-image models. arXiv preprint arXiv:2605.21573 (2026)

[9] Chen, H., Li, W., Gu, J., Ren, J., Chen, S., Ye, T., Pei, R., Zhou, K., Song, F., Zhu, L.: RestoreAgent: Autonomous image restoration agent via multimodal large language models. In: Advances in Neural Information Processing Systems. vol. 37, pp. 110643–110666 (2024)

[10] Chen, J., Yu, J., Ge, C., Yao, L., Xie, E., Wu, Y., Wang, Z., Kwok, J., Luo, P., Lu, H., Li, Z.: PixArt-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In: International Conference on Learning Representations (2024)

[11] Chen, S., Lai, J., Gao, J., Shi, H., Liu, Z., Ye, T., Luo, J., Wei, X., Zhu, L.: PosterOmni: Generalized artistic poster creation via task distillation and unified reward feedback. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5978–5987 (2026)

[12] Chen, S., Lai, J., Gao, J., Ye, T., Chen, H., Shi, H., Shao, S., Lin, Y., Fei, S., Xing, Z., Jin, Y., Luo, J., Wei, X., Zhu, L.: PosterCraft: Rethinking high-quality aesthetic poster generation in a unified framework. In: International Conference on Learning Representations (2026)

[13] Chen, S., Ye, T., Lin, Y., Jin, Y., Yang, Y., Chen, H., Lai, J., Fei, S., Xing, Z., Tsung, F., Zhu, L.: GenHaze: Pioneering controllable one-step realistic haze generation for real-world dehazing. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9194–9205 (2025)

[14] Chen, X., Wang, X., Zhou, J., Qiao, Y., Dong, C.: Activating more pixels in image super-resolution transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22367–22377 (2023)

[15] Cohen, R., Kligvasser, I., Rivlin, E., Freedman, D.: Looks too good to be true: An information-theoretic analysis of hallucinations in generative restoration models. Advances in Neural Information Processing Systems **37**, 22596–22623 (2024)

[16] Dang-Nguyen, D.T., Pasquini, C., Conotter, V., Boato, G.: RAISE: A raw images dataset for digital image forensics. In: Proceedings of the 6th ACM Multimedia Systems Conference. pp. 219–224. MMSys '15, Association for Computing Machinery, New York, NY, USA (2015). https://doi.org/10.1145/2713168.2713194

[17] DeepSeek-AI: DeepSeek-V4: Towards highly efficient million-token context intelligence. arXiv preprint arXiv:2606.19348 (2026)

[18] Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: Proceedings of the European Conference on Computer Vision. pp. 184–199. Springer (2014)

[19] Duan, Z.P., Zhang, J., Jin, X., Zhang, Z., Xiong, Z., Zou, D., Ren, J., Guo, C.L., Li, C.: DiT4SR: Taming diffusion transformer for real-world image super-resolution. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2025)

[20] Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al.: Scaling rectified flow transformers for high-resolution image synthesis. In: Forty-first International Conference on Machine Learning (2024)

[21] Fei, S., Wang, G., Ye, T., Chen, S., Lai, J., Zhu, L.: S2DLight: Single step diffusion network via degradation-aware adaptation for surgical endoscopic image low-light enhancement. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 444–454 (2026)

[22] Fei, S., Ye, T., Wang, L., Zhu, L.: LucidFlux: Caption-free photo-realistic image restoration via a large-scale diffusion transformer. In: International Conference on Learning Representations (2026)

[23] Fleiss, J.L.: Measuring nominal scale agreement among many raters. Psychological Bulletin **76**(5), 378–382 (1971)

[24] Gao, Y., Guo, H., Hoang, T., Huang, W., Jiang, L., et al.: Seedance 1.0: Exploring the boundaries of video generation models. arXiv preprint arXiv:2506.09113 (2025)

[25] Giakoumoglou, P., Karageorgiou, D., Papadopoulos, S., Petrantonakis, P.C.: SAGI: Semantically aligned and uncertainty guided AI image inpainting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2025)

[26] Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: LoRA: Low-rank adaptation of large language models. In: International Conference on Learning Representations (2022)

[27] Huang, S., Liu, S., Wei, Z.: LPNSR: Prior-enhanced diffusion image super-resolution via LR-Guided noise prediction. arXiv preprint arXiv:2603.21045 (2026)

[28] Huang, Y., Zhou, X., Zhang, J., Bo, L., Pang, T.: MeanFlowNFT: Bringing forward-process RL to average-velocity generators. arXiv preprint arXiv:2607.15273 (2026)

[29] Ju, X., Liu, X., Wang, X., Bian, Y., Shan, Y., Xu, Q.: BrushNet: A plug-and-play image inpainting model with decomposed dual-branch diffusion. In: European Conference on Computer Vision (2024)

[30] Ke, J., Wang, Q., Wang, Y., Milanfar, P., Yang, F.: MUSIQ: Multi-scale image quality transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5148–5157 (2021)

[31] Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., Krishnan, D.: Supervised contrastive learning. In: Advances in Neural Information Processing Systems. vol. 33, pp. 18661–18673 (2020)

[32] Lai, J., Chen, S., Gao, J., Shi, H., Liu, Z., Zhai, F., Luo, J., Wei, X., Wang, L., Zhu, L.: PosterReward: Unlocking accurate evaluation for high-quality graphic design generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7762–7772 (2026)

[33] Lai, J., Chen, S., Lin, Y., Ye, T., Liu, Y., Fei, S., Xing, Z., Wu, H., Wang, W., Zhu, L.: SnowMaster: Comprehensive real-world image desnowing via MLLM with multi-model feedback optimization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4302–4312 (2025)

[34] Landis, J.R., Koch, G.G.: The measurement of observer agreement for categorical data. Biometrics **33**(1), 159–174 (1977)

[35] Li, M., Zhang, Y., Long, D., Chen, K., Song, S., Bai, S., Yang, Z., Xie, P., Yang, A., Liu, D., et al.: Qwen3-VL-Embedding and Qwen3-VL-Reranker: A unified framework for state-of-the-art multimodal retrieval and ranking. arXiv preprint arXiv:2601.04720 (2026)

[36] Li, X., Xing, Z., Liu, Z., Li, S., Wang, H., Fei, S., Zhu, L.: MoSE: Mixture-of-scale-experts for medical image restoration. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 306–316 (2026)

[37] Li, Y., Zhang, K., Liang, J., Cao, J., Liu, C., Gong, R., Zhang, Y., Tang, H., Liu, Y., Demandolx, D., Ranjan, R., Timofte, R., Van Gool, L.: LSDIR: A large scale dataset for image restoration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1775–1787 (2023)

[38] Li, Z., Liu, Z., Zhang, Q., Lin, B., Wu, F., Yuan, S., Yan, Z., Ye, Y., Yu, W., Niu, Y., Wang, S., Cheng, X., Yuan, L.: UniWorld-V2: Reinforce image editing with diffusion negative-aware finetuning and MLLM implicit feedback. arXiv preprint arXiv:2510.16888 (2025)

[39] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common objects in context. In: European Conference on Computer Vision. pp. 740–755 (2014)

[40] Lin, X., He, J., Chen, Z., Lyu, Z., Dai, B., Yu, F., Qiao, Y., Ouyang, W., Dong, C.: DiffBIR: Toward blind image restoration with generative diffusion prior. In: European Conference on Computer Vision. pp. 430–448. Springer (2024)

[41] Lipman, Y., Chen, R.T., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. In: International Conference on Learning Representations (2023)

[42] Liu, J., Liu, G., Liang, J., Li, Y., Liu, J., Wang, X., Wan, P., Zhang, D., Ouyang, W.: Flow-GRPO: Training flow matching models via online RL. In: Advances in Neural Information Processing Systems (2025)

[43] Liu, X., Gong, C., Liu, Q.: Flow straight and fast: Learning to generate and transfer data with rectified flow. arXiv preprint arXiv:2209.03003 (2022)

[44] Liu, Y., Fan, Z., Huang, J., Xiao, J., Li, D., Zhang, W., Bai, L., Fu, X., Zha, Z.J.: FinPercep-RM: A fine-grained reward model and co-evolutionary curriculum for RL-based real-world super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4839–4849 (2026)

[45] Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: International Conference on Learning Representations (2019)

[46] Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C., Zhu, J.: DPM-Solver: A fast ODE solver for diffusion probabilistic model sampling in around 10 steps. In: Advances in Neural Information Processing Systems. vol. 35, pp. 5775–5787 (2022)

[47] Manukyan, H., Sargsyan, A., Atanyan, B., Wang, Z., Navasardyan, S., Shi, H.: HD-Painter: High-resolution and prompt-faithful text-guided image inpainting with diffusion models. In: International Conference on Learning Representations (2025)

[48] Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)

[49] Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al.: Training language models to follow instructions with human feedback. In: Advances in Neural Information Processing Systems (2022)

[50] Ping, B., Jia, C., Luo, M., Qian, H., Tsang, I.: Flow-Factory: A unified framework for reinforcement learning in flow-matching models. arXiv preprint arXiv:2602.12529 (2026)

[51] Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., Rombach, R.: SDXL: Improving latent diffusion models for high-resolution image synthesis. In: International Conference on Learning Representations (2024)

[52] Qiao, J., Cai, M., Li, W., Liu, Y., Huang, X., He, G., Xie, J., Hu, J., Chen, X., Lin, S.: RealSR-R1: Reinforcement learning for real-world image super-resolution with vision-language chain-of-thought. In: arXiv preprint arXiv:2506.16796 (2025)

[53] Qu, Y., Yuan, K., Zhao, K., Xie, Q., Hao, J., Sun, M., Zhou, C.: XPSR: Cross-modal priors for diffusion-based image super-resolution. In: European Conference on Computer Vision. pp. 285–303 (2024)

[54] Rafailov, R., Sharma, A., Mitchell, E., Manning, C.D., Ermon, S., Finn, C.: Direct preference optimization: Your language model is secretly a reward model. Advances in neural information processing systems **36**, 53728–53741 (2023)

[55] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10684–10695 (2022)

[56] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347 (2017)

[57] Song, Y., Quan, W., Wang, W., Sun, J., Liu, J., Li, M., Yu, P., Chen, Z., Shen, W., Yuan, L., et al.: RefReward-SR: LR-Conditioned reward modeling for preference-aligned super-resolution. arXiv preprint arXiv:2603.24198 (2026)

[58] Super Intelligence Team: FireRed-Image-Edit-1.0 technical report. arXiv preprint arXiv:2602.13344 (2026)

[59] Sutton, R.S., Barto, A.G., et al.: Reinforcement learning: An introduction, vol. 1. MIT Press (1998)

[60] Talebi, H., Milanfar, P.: NIMA: Neural image assessment. IEEE Transactions on Image Processing **27**(8), 3998–4011 (2018)

[61] Wallace, B., Dang, M., Rafailov, R., Zhou, L., Lou, A., Purushwalkam, S., Ermon, S., Xiong, C., Joty, S., Naik, N.: Diffusion model alignment using direct preference optimization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8228–8238 (2024)

[62] Wang, J., Chan, K.C., Loy, C.C.: Exploring CLIP for assessing the look and feel of images. In: AAAI (2023)

[63] Wang, J., Yue, Z., Zhou, S., Chan, K.C., Loy, C.C.: Exploiting diffusion prior for real-world image super-resolution. International Journal of Computer Vision **132**(12), 5929–5949 (2024)

[64] Wang, X., Xie, L., Dong, C., Shan, Y.: Real-ESRGAN: Training real-world blind super-resolution with pure synthetic data. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 1905–1914 (2021)

[65] Wang, Y., Yang, W., Chen, X., Wang, Y., Guo, L., Chau, L.P., Liu, Z., Qiao, Y., Kot, A.C., Wen, B.: 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)

[66] Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing **13**(4), 600–612 (2004)

[67] Wei, P., Xie, Z., Lu, H., Zhan, Z., Ye, Q., Zuo, W., Lin, L.: Component divide-and-conquer for real-world image super-resolution. In: Proceedings of the European Conference on Computer Vision. pp. 101–117. Springer (2020)

[68] Wu, H., Zhang, Z., Zhang, W., Chen, C., Li, C., Liao, L., Wang, A., Zhang, E., Sun, W., Yan, Q., Min, X., Zhai, G., Lin, W.: Q-Align: Teaching LMMs for visual scoring via discrete text-defined levels. In: International Conference on Machine Learning (2024)

[69] Wu, R., Sun, L., Zhang, Z., Wang, S., Wu, T., Yi, Q., Li, S., Zhang, L.: DP^{2}O-SR: Direct perceptual preference optimization for real-world image super-resolution. In: Advances in Neural Information Processing Systems (2025)

[70] Wu, R., Yang, T., Sun, L., Zhang, Z., Li, S., Zhang, L.: SeeSR: Towards semantics-aware real-world image super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 25456–25467 (2024)

[71] Wu, T., Zou, J., Liang, J., Zhang, L., Ma, K.: VisualQuality-R1: Reasoning-induced image quality assessment via reinforcement learning to rank. In: Advances in Neural Information Processing Systems (2025)

[72] Xue, Z., Wu, J., Gao, Y., Kong, F., Zhu, L., Chen, M., Liu, Z., Liu, W., Guo, Q., Huang, W., et al.: DanceGRPO: Unleashing GRPO on visual generation. arXiv preprint arXiv:2505.07818 (2025)

[73] Yang, S., Wu, T., Shi, S., Lao, S., Gong, Y., Cao, M., Wang, J., Yang, Y.: MANIQA: Multi-dimension attention network for no-reference image quality assessment. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1191–1200 (2022)

[74] Ye, T., Chen, S., Chai, W., Xing, Z., Qin, J., Lin, G., Zhu, L.: Learning diffusion texture priors for image restoration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2524–2534 (2024)

[75] Ye, T., Fei, S., Zhu, L.: UltraFlux: Data-model co-design for high-quality native 4K text-to-image generation across diverse aspect ratios. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22070–22079 (2026)

[76] Yi, Q., Li, S., Wu, R., Sun, L., Zhang, Z., Zhang, L.: GDPO-SR: Group direct preference optimization for one-step generative image super-resolution. arXiv preprint arXiv:2603.16769 (2026)

[77] Yu, F., Gu, J., Li, Z., Hu, J., Kong, X., Wang, X., He, J., Qiao, Y., Dong, C.: Scaling up to excellence: Practicing model scaling for photo-realistic image restoration in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 25669–25680 (2024)

[78] Yu, T., Feng, R., Feng, R., Liu, J., Jin, X., Zeng, W., Chen, Z.: Inpaint anything: Segment anything meets image inpainting. arXiv preprint arXiv:2304.06790 (2023)

[79] Yue, Z., Wang, J., Loy, C.C.: ResShift: Efficient diffusion model for image super-resolution by residual shifting. Advances in Neural Information Processing Systems **36**, 13294–13307 (2023)

[80] Zhang, L., Zhang, L., Bovik, A.C.: A feature-enriched completely blind image quality evaluator. IEEE Transactions on Image Processing **24**(8), 2579–2591 (2015)

[81] Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3836–3847 (2023)

[82] Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: Proceedings of the European Conference on Computer Vision. pp. 286–301 (2018)

[83] Zhao, S., Zhang, X., Chen, B., Li, W., Xing, Q., Zhang, K., Wang, Y., Li, J., Zhang, L., Zhang, J., et al.: OARS: Process-aware online alignment for generative real-world image super-resolution. arXiv preprint arXiv:2603.12811 (2026)

[84] Zheng, K., Chen, H., Ye, H., Wang, H., Zhang, Q., Jiang, K., Su, H., Ermon, S., Zhu, J., Liu, M.Y.: DiffusionNFT: Online diffusion reinforcement with forward process. In: International Conference on Learning Representations (2026)

[85] Zhuang, J., Zeng, Y., Liu, W., Yuan, C., Chen, K.: A task is worth one word: Learning with task prompts for high-quality versatile image inpainting. In: European Conference on Computer Vision (2024)

A Additional Related Work

Preference and reward-based Real-ISR alignment. Recent works have explored preference or reward optimization specifically for Real-ISR. DSPO [6] applies semantic preference optimization, DP^{2}O-SR [69] studies direct perceptual preference optimization, GDPO-SR [76] introduces group DPO for one-step generative ISR, and OARS [83] performs process-aware online alignment with an MLLM-based reward. RefReward-SR [57] and FinPercep-RM [44] further study LR-conditioned or fine-grained reward modeling.

DP^{2}O-SR is the most closely related preference-based Real-ISR work to our setting. It constructs hybrid IQA rewards, selects winner–loser pairs from stochastic SR candidates, and optimizes a DPO-style hierarchical preference objective. LucidNFT targets a different bottleneck in flow-matching Real-ISR: LR-referenced faithfulness is explicitly modeled by LucidConsistency, and perceptual and faithfulness rewards are normalized separately within each LR-conditioned rollout group before fusion. This preserves objective-wise reward contrast, rather than collapsing heterogeneous objectives into pairwise scalar preferences.

A direct comparison with DP^{2}O-SR is not feasible: its public release provides inference scripts and checkpoints for C-SD2/C-FLUX, but not the training code, training data, or IQA reward labels required to reproduce its post-training pipeline. In addition, its released flow-based artifact follows a ControlNet-FLUX-style pipeline, whereas our controlled experiments use LucidFlux as the base model. A direct DP^{2}O-SR-on-LucidFlux comparison would therefore require a substantial re-implementation of rollout generation, IQA labeling, pair selection, and HPO-weighted training. We instead report a same-backbone Diffusion-DPO [61] implementation, a form of preference optimization also used in recent generation pipelines [12], to isolate direct preference optimization under our data, backbone, and training setting.

Learned evaluators and reward feedback. Beyond Real-ISR, learned evaluators and reward feedback have been used to steer generative and restoration models toward task-specific criteria. SnowMaster [33] uses an MLLM evaluator refined by multi-model feedback optimization to assess real-world desnowing results and filter pseudo-labels, PosterReward [32] models the typography and layout quality of graphic designs, and PosterOmni [11] trains a unified reward model to guide reinforcement learning for poster creation. LucidConsistency plays the corresponding role for LR faithfulness in Real-ISR, where the reference is a degraded input rather than a text prompt.

LR-guided generative super-resolution. Several non-RL approaches improve LR anchoring through model design. XPSR [53] introduces cross-modal priors for diffusion-based SR, LPNSR [27] strengthens LR-guided noise prediction to reduce hallucination, and related restoration models learn dedicated diffusion texture priors [74]. These methods modify the restoration prior or conditioning mechanism, whereas LucidNFT aligns stochastic SR rollouts through reward-guided fine-tuning; as it leaves the backbone architecture unchanged, it could also be applied to stronger flow priors, such as FLUX-based models scaled to native 4K generation [75].

Real-world image super-resolution datasets. Training and evaluating Real-ISR models requires data that reflect realistic degradation patterns. Synthetic pipelines such as Real-ESRGAN [64] generate blur, noise, and compression artifacts from high-quality datasets such as DIV2K, Flickr2K [1], and LSDIR [37], and generative models can synthesize more realistic degradations such as haze [13]. Real-captured paired datasets such as RealSR [5] and DRealSR [67] are valuable benchmarks, but their scale and capture diversity are limited. LucidLR is designed for a different role: it provides large-scale unpaired real LR inputs for RL fine-tuning, where diverse degradations are needed to induce informative rollout variation and reward contrast. Specialized domains such as surgical endoscopy [21] and medical imaging [36] involve degradations that web photographs do not cover, which motivates limitation (iii) in Sec. 6.

B LucidConsistency: Data, Human Alignment, and Robustness

B.1 Training Data Construction

Content-consistent pools. We generate degradation views using a controlled Real-ESRGAN-style pipeline. For each HR crop, we produce three ordered degradation levels: mild, medium, and heavy. The clean crop is saved as the GT view, while each degraded LR image is bicubic-upsampled back to the GT resolution to form the corresponding degraded view of the content-consistent pool.

Original–inpainted pairs. For hard negatives, we use the fully regenerated subset of SAGI [25]. Each sample consists of an original image and its inpainted counterpart. SAGI inpainting methods output images at their own working resolution (typically around 1K pixels on the long side), which can differ substantially from that of the original photograph (up to about 5K pixels for RAISE). We therefore first bring each pair to a common pixel grid: the original is resized to the resolution of its inpainted counterpart, so that both images depict the same scene at the same scale, and the original is never compared against an upsampled, artificially smoothed inpainted image. If the short side of the pair is below the crop size, both images are bicubically upscaled by the same factor rather than padded. Both images then share the same random crop coordinates within each epoch, and the crop is sampled so that it contains the inpainted region given by the SAGI edit mask; the paired crops are therefore identical outside the edit and differ only in the locally regenerated content. Their degradation operations are sampled independently, so the paired views remain spatially aligned while covering different blur, noise, resizing, and compression realizations.

Cropping. All HR crops use size 1024×1024. For each source image we generate four crops (one per epoch), each at the three degradation levels. Different epochs use different crop regions, while paired original–inpainted samples share crop coordinates within the same epoch. Content-consistent pools follow the same rule for small images: images whose short side is below 1024 pixels are bicubically upscaled before cropping instead of being padded.

Degradation levels. Each degradation level randomly samples Real-ESRGAN operations from a predefined range:

- **Mild**: weak blur, mostly keeping the original resolution or slight downsampling, very light Gaussian/Poisson noise, high JPEG quality, and no second-order degradation.
- **Medium**: moderate blur, stronger downsampling, moderate noise, medium JPEG compression, and occasional second-order degradation.
- **Heavy**: strong blur, aggressive downsampling, stronger noise, lower JPEG quality, and always applying second-order degradation.

Table 6 lists the parameter ranges.

Table 6: Parameter ranges of the three Real-ESRGAN degradation levels used to build content-consistent pools.

Parameter	Mild	Medium	Heavy
Blur sigma	$[0.2, 0.45]$	$[0.6, 1.5]$	$[1.2, 3.0]$
Resize range	$[0.9, 1.0]$	$[0.5, 0.8]$	$[0.25, 0.6]$
Noise range	$[0, 1.5]$	$[3, 8]$	$[8, 20]$
Poisson scale	$[0.0, 0.12]$	$[0.3, 1.0]$	$[1.0, 2.5]$
JPEG quality	$[92, 100]$	$[50, 85]$	$[20, 60]$
Second-order prob.	0.0	0.2	1.0

The pipeline follows the Real-ESRGAN two-stage degradation design with blur, resizing, noise, JPEG compression, and sinc filtering. For the medium and heavy levels, the second-stage degradation uses the same noise and JPEG ranges as the first stage, while the second-stage resize ranges are $[0.5, 0.8]$ and $[0.25, 0.8]$, respectively.

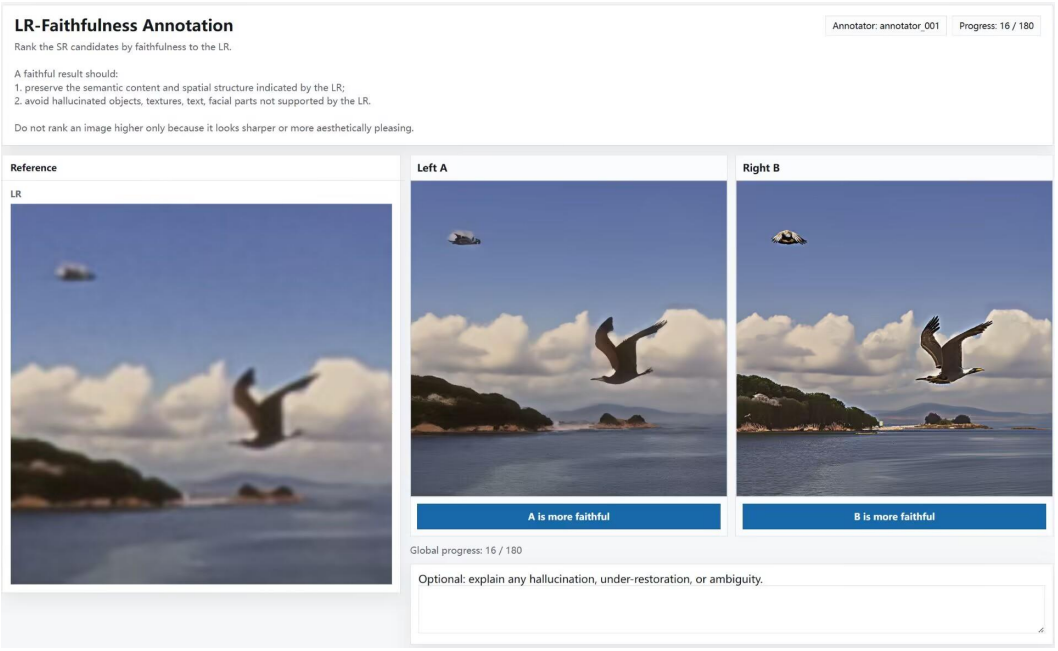

Figure 5: Annotation interface for LR-faithfulness evaluation. Annotators are shown the LR image and four randomly ordered SR candidates, and rank the candidates by faithfulness to the LR evidence rather than perceptual sharpness alone.

B.2 Human Alignment Protocol

We provide additional details for the human-aligned LR-faithfulness evaluation in Sec. 5.2. The study is conducted on RealSR [5]. For each LR image, we collect SR candidates from eight generative SR methods: ResShift [79], SeeSR [70], LucidFlux [22], DreamClear [2], SUPIR [77], DiffBIRv2 [40], DiT4SR [19], and StableSR [63]. Each annotation task presents the LR image and four randomly sampled SR candidates from this candidate pool.

Five expert annotators are asked to judge LR-faithfulness rather than perceptual sharpness or aesthetics. Specifically, annotators are instructed to rank candidates according to whether the restored result preserves LR-supported semantic content and spatial structure, while avoiding hallucinated objects, textures, text, facial parts, or repeated patterns unsupported by the LR observation. The annotation interface is shown in Fig. 5. Each four-candidate ranking is decomposed into six pairwise preferences, and candidate order is randomized to reduce presentation bias.

We evaluate automatic metrics by comparing their induced rankings with the aggregated human preferences. We report three metrics: **Agreement**, the pairwise agreement with human preferences; **Recall@1**, whether the metric selects the human-preferred best candidate; and **Filter@1**, whether the metric assigns the lowest score to the human-labeled worst candidate. As reported in the main paper, LucidConsistency achieves higher agreement with human LR-faithfulness judgments than perceptual metrics and the frozen Qwen3-VL-Embedding baseline.

B.3 Leave-One-Inpainter-Out Evaluation

Protocol. Our hard-negative pool contains 11 single-round combinations of four inpainting methods and seven checkpoints from the SD 1.5, SD 2, and SDXL families, plus two-round edits (Sec. 5.1). The SDXL family is used only by BrushNet, with the SDXL base model and with Juggernaut XL. Removing every pair in which either of these two generators takes part, in either editing round (2,784 of the 37.7K pairs: 1,437 single-round and 1,347 two-round), therefore withholds an entire base architecture, which is the strongest single-fold test of architectural generalization available in our training pool. We retrain LucidConsistency on the remaining pairs under otherwise identical settings (LOIO) and compare it with the evaluator trained on all pairs (Full).

Test sets. (i) *SAGI test split of the held-out SDXL-family generators*, disjoint from training. For an original image x^o with such an inpainting x^i, a faithful evaluator should score the content-preserving pair $(x^o, D(x^o))$ above the content-altered pair (x^o, x^i), where $D(\cdot)$ is a Real-ESRGAN synthetic degradation at a mild, medium, or heavy level. (ii) *Human-annotated set*: the 40 RealSR LR inputs of Appendix D.1, each paired with the two real SR outputs, with the human majority vote as ground truth. On both sets we report paired agreement, i.e., the fraction of pairs on which the evaluator's preference matches the ground truth.

Table 7: Paired agreement of LucidConsistency with and without the SDXL-family generators in training.

Evaluator	**SAGI test split ↑**	**Human-annotated set ↑**
Full (all generators)	0.74	0.69
LOIO (SDXL family held out)	0.70	0.67

C LucidNFT: Implementation and Training Dynamics

C.1 Hyperparameters and Pseudocode

Table 8 lists the full LucidNFT training configuration used for both backbones in the main experiments, and Algorithm C.1 summarizes one training iteration. Relative to DiffusionNFT [84], only the mapping from the reward vector to the reward weight (step 3) differs; the policy objective is unchanged.

Table 8: LucidNFT training hyperparameters.

Item	**Setting**
Hardware / RL steps	8 NVIDIA A100 GPUs / 600 steps
Rollout resolution	768×768
LR groups	3 per GPU (24 globally)
Rollouts per LR input (M)	12
Rewards and weights	UniPercept IQA ($\lambda_p = 1.0$), LucidConsistency ($\lambda_f = 1.0$)
LoRA rank / alpha	32 / 64
Optimizer	AdamW, lr 3×10^{-5}, $\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay 10^{-4}
NFT interpolation weight β_{NFT}	1.0
KL regularization weight λ_{KL}	10^{-4}
Old-policy mixing η_t (Alg. C.1)	$\min(10^{-3}t,\ 0.5)$
EMA of policy weights (evaluation)	decay 0.9
Z_c (advantage clip) / grad-norm clip	5.0 / 1.0
Timestep-training fraction	0.99
Update epochs per rollout batch	1 (no rollout replay)
Sampler	DPM-Solver-2 [46], 6 steps (training) / 15 steps (validation)
Rollout noise level	0.7 (deterministic after injection)
Guidance scale / precision	4 / bf16 (matched to LucidFlux)

Algorithm C.1 (One LucidNFT Training Iteration)

Input: policy v_θ, sampling policy v_{old} (no gradient), rewards $\{R^{(k)}\}_{k=1}^{K}$ with weights $\{\lambda_k\}$, LR batch $\{x_i\}$.

1. **Rollout.** For each LR input x_i, sample M restorations $\{y_{i,j}\}_{j=1}^{M}$ from v_{old} with the black-box solver.
2. **Score.** Compute the reward vector $\mathbf{r}_{i,j} = (R^{(1)}(x_i, y_{i,j}), \ldots, R^{(K)}(x_i, y_{i,j}))$.
3. **Decoupled normalization.** Normalize each reward within its rollout group (Eq. (10)), fuse $\tilde{a}_{i,j} = \sum_k \lambda_k z_{i,j}^{(k)}$, standardize over the batch, clip, and map to $r_{i,j}^{L} \in [0, 1]$ (Eq. (11)).
4. **NFT update.** Sample t and noise, form x_t from $y_{i,j}$, and minimize Eq. (4) with $r = r_{i,j}^{L}$, plus the KL term weighted by λ_{KL}; one epoch over the fresh rollouts.
5. **Old-policy update.** Set $v_{\text{old}} \leftarrow \eta_t v_{\text{old}} + (1 - \eta_t) v_\theta$ with $\eta_t = \min(10^{-3}t, 0.5)$.

C.2 Advantage Separability

Fig. 2c counts how many distinct advantage levels a group contains. Fig. 6 complements it with the size of the contrast: for each group of the same 16 LR inputs, we take the two candidates whose rewards differ most and measure the gap between their advantages. Scalar-first fusion keeps this gap small, 0.3–0.7, because candidates that win on one reward and lose on the other obtain similar sums; decoupled normalization makes it three to four times larger for every group size (1.27 vs. 0.34 at $M = 2$, 2.19 vs. 0.72 at $M = 12$) and larger in 12 of the 16 individual groups, so the most different candidates receive markedly different update weights.

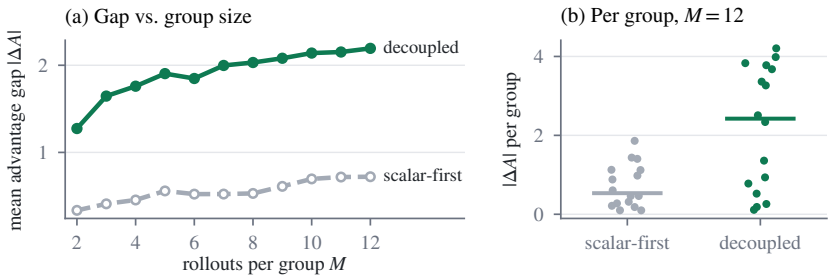

Figure 6: Advantage gap $|\Delta A|$ between the two candidates with the largest reward difference in a group (16 LR inputs, 12 LucidFlux rollouts each, same data as Fig. 2). (a) Mean over groups versus the number of rollouts M. (b) Individual groups at $M = 12$; bars mark the median.

C.3 Training Dynamics

Fig. 7 shows the training dynamics of LucidNFT and Diffusion-DPO. Validation rewards are computed with fewer sampling steps than the final evaluation, and LucidConsistency varies less than the perceptual reward during training; the final evaluation in Table 4 shows that LucidNFT improves LucidConsistency over the corresponding backbones. Fig. 8 compares three reward formulations from Table 3. IQA-only RL (A) raises the perceptual reward above the pretrained baseline; adding LucidConsistency through scalar-first fusion (C) lowers it again, reflecting the perceptual–faithfulness trade-off; and decoupled normalization (E) yields the highest perceptual reward of the three, consistent with the controlled ablation.

D Additional Results

D.1 Model-Level Human Preference Study

The study in Sec. 5.3 compares trained models directly, without any LucidConsistency scores. We use 40 held-out RealSR [5] LR inputs. For each input, the outputs of LucidFlux and LucidFlux(+LucidNFT) are shown side by side with the LR image, in random order. Five expert annotators are asked to select the output that is better supported by the LR content, explicitly disregarding sharpness, color saturation, and aesthetic quality; ties are not allowed. Every annotator judges all 40 pairs, giving 200 binary judgments.

Table 9: Model-level human preference on LR faithfulness (RealSR).

Benchmark	Annotators	Judgments	LucidFlux(+LucidNFT)	LucidFlux	Fleiss' κ
RealSR	5	200	**72.5%**	27.5%	0.415

A Fleiss' κ [23] of 0.415 corresponds to moderate agreement on the Landis–Koch scale [34], indicating that the preference reflects consensus across independent annotators rather than a narrow-majority effect.

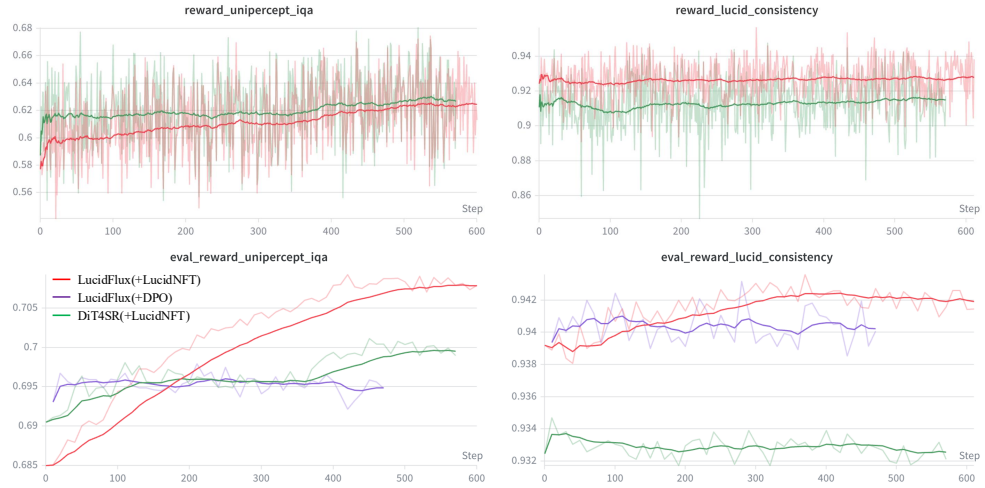

Figure 7: Training curves of LucidNFT and DPO fine-tuning. Curves report fast validation rewards computed with reduced inference steps. LucidNFT steadily improves perceptual rewards while keeping LucidConsistency stable; the final full evaluation in Table 4 shows improved LR-faithfulness over the corresponding backbones.

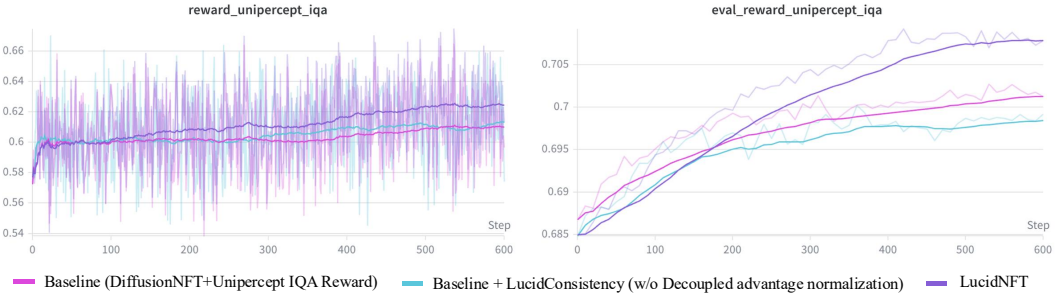

Figure 8: UniPercept IQA reward during training for three LucidFlux variants of Table 3: IQA-only RL (A), scalar-first fusion with LucidConsistency (C), and decoupled normalization (E).

D.2 Relative Improvements over the LucidFlux Backbone

Table 10 reports the relative change of LucidFlux(+DPO) and LucidFlux(+LucidNFT) over the LucidFlux backbone for every metric and benchmark of Table 4. For NIQE, a negative change denotes improvement.

Table 10: Relative change (%) over the same LucidFlux backbone. DPO denotes Diffusion-DPO; UP-IQA denotes UniPercept IQA; LC denotes LucidConsistency.

Benchmark	Method	CLIP-IQA+↑	Q-Align↑	MUSIQ↑	MANIQA↑	CLIP-IQA↑	NIQE↓	UP-IQA↑	VQ-R1↑	LC↑
RealLQ250	+DPO	+0.28	+0.86	+0.16	+0.59	+0.90	+1.00	+0.29	+0.37	+0.64
	+LucidNFT	+3.57	+1.82	+1.54	+4.13	+5.51	−13.04	+3.59	+2.28	+1.17
DRealSR	+DPO	+0.21	+0.72	−0.07	−0.19	+0.60	−1.18	−0.21	−0.32	+0.87
	+LucidNFT	+5.39	+3.45	+5.50	+6.97	+5.99	−17.65	+6.47	+3.75	+0.75
RealSR	+DPO	+0.39	+1.08	+0.06	+0.37	+1.07	−2.76	+0.64	+0.31	+0.89
	+LucidNFT	+7.23	+3.07	+3.93	+8.08	+9.07	−17.88	+7.77	+4.87	+0.79
Average	+DPO	+0.29	+0.89	+0.05	+0.26	+0.86	−0.98	+0.24	+0.12	+0.80
	+LucidNFT	+5.39	+2.78	+3.66	+6.39	+6.87	−16.19	+5.94	+3.63	+0.90

D.3 Robustness to Reward Weights and Reward Sets

We train ten configurations on LucidLR for 300 RL steps with all other hyperparameters as in Table 8, and evaluate on RealLQ250: five reward setups, each with scalar-first fusion and with decoupled normalization. The setups are three weightings $\lambda_p{:}\lambda_f \in \{1{:}1, 2{:}1, 1{:}2\}$ of UniPercept IQA (UP-

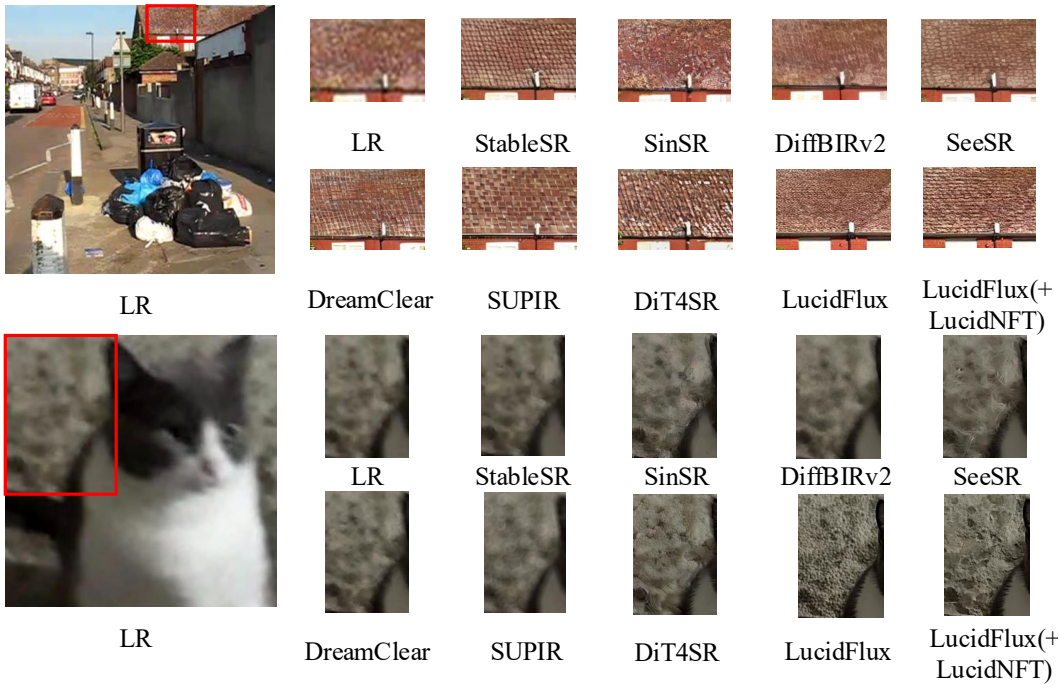

Figure 9: Additional qualitative comparisons on RealLQ250 [2]. Compared with the LucidFlux baseline, LucidFlux(+LucidNFT) recovers clearer local details while better preserving LR-supported structures.

IQA) and LucidConsistency (LC), and two three-reward extensions that add a perceptual reward (CLIP-IQA) or a semantic reward (Qwen3-VL-Sim, the cosine similarity between LR and SR under the frozen Qwen3-VL-Embedding-8B backbone). All ten runs share the same budget, isolating the relative behavior of the two fusion schemes. "Decoupled wins" counts, per configuration, the metrics on which decoupled normalization yields the larger gain; it summarizes directional consistency across correlated metrics rather than independent statistical tests.

Table 11: Reward-configuration grid on RealLQ250. The first row is absolute; other rows report changes from the backbone (larger is better on all columns).

Configuration	Fusion	UP-IQA↑	CLIP-IQA↑	CLIP-IQA+↑	MUSIQ↑	MANIQA↑	LC↑	Decoupled wins
LucidFlux backbone	–	70.93	0.686	0.721	72.34	0.523	0.9237	–
UP-IQA : LC = 1:1	scalar-first	+1.21	+0.016	+0.005	+0.02	−0.002	+0.0144	6/6
	decoupled	**+1.69**	**+0.028**	**+0.015**	**+0.55**	**+0.011**	**+0.0170**	
UP-IQA : LC = 2:1	scalar-first	+1.01	+0.009	+0.002	−0.03	−0.006	**+0.0183**	5/6
	decoupled	**+1.70**	**+0.027**	**+0.016**	**+0.69**	**+0.013**	+0.0157	
UP-IQA : LC = 1:2	scalar-first	+0.78	+0.009	−0.002	−0.34	−0.010	**+0.0175**	5/6
	decoupled	**+0.92**	**+0.018**	**+0.005**	**+0.10**	**+0.002**	+0.0169	
UP-IQA : LC : CLIP-IQA = 1:1:1	scalar-first	+0.46	+0.004	−0.005	−0.43	−0.013	+0.0163	6/6
	decoupled	**+1.31**	**+0.027**	**+0.011**	**+0.39**	**+0.011**	**+0.0176**	
UP-IQA : LC : Qwen3-VL-Sim = 1:1:1	scalar-first	+0.91	+0.010	−0.004	−0.48	−0.010	+0.0163	6/6
	decoupled	**+0.95**	**+0.016**	**+0.004**	**+0.15**	**+0.005**	**+0.0172**	

D.4 Additional Qualitative Results

Fig. 9 shows further RealLQ250 examples of LucidFlux and LucidFlux(+LucidNFT). The fine-tuned model restores sharper local structures, such as text strokes and regular patterns, while keeping them consistent with the LR input; these examples are illustrative, and the quantitative and human evaluations in Sec. 5.3 remain the primary evidence.

E LucidLR Dataset

E.1 Collection, Screening, and Release

Collection. Images are retrieved from the Wikimedia Commons categories *Images of low quality* and *Blurred images* through the official API (`https://commons.wikimedia.org`). The raw pool contains 21,698 unique files.

Content screening. Every image is screened by DeepSeek-V4.1-Flash at temperature 0, after downscaling (never upscaling) so that the longer side is at most 1024 pixels. A first pass labels each image as safe or as one of *nudity/sexual*, *suggestive*, *gore/violence*, *drugs*, *hate symbols*, or *other*. Images flagged by the first pass are re-examined by a narrower second pass that separates photographic content from depictions in art (e.g., classical nudity or religious and battle paintings) and corrects false alarms such as dark scenes. Table 12 lists the resulting filtering steps; artistic depictions are kept.

Table 12: Filtering steps from the collected pool to the released LucidLR.

Step	**Images**
Pool after de-duplication, excluding a 1,000-image held-out split	20,698
− explicit sexual content, photographic gore, personal data	−56
− hate symbols	−16
− short side below 128 pixels	−719
− source file deleted from Commons (license unverifiable)	−169
+ screened images collected from the same categories	+262
Released LucidLR	**20,000**

Licensing. LucidLR includes only images whose Commons licenses permit redistribution. Of the 20,000 images, 12,542 are under CC BY-SA (2.0/2.5/3.0/4.0 and ported variants), 2,780 under CC BY, 4,469 are public domain, CC0, or otherwise free of restrictions, and 209 carry other free licenses (e.g., GFDL or Free Art License). Any image whose redistribution status is uncertain (e.g., trademark, personality-rights, or freedom-of-panorama concerns) is excluded from the image bundle and kept as a URL-only entry in the metadata index, so that users can still reproduce the collection while responsibility for local redistribution remains with them.

Release format. We release LucidLR in two forms: (1) a pre-fetched image bundle, since the Commons API enforces per-user rate limits that make re-downloading 20K images slow; and (2) a metadata index with a reference downloader, `download_lucidlr.py`, for users who prefer to fetch the images themselves. Every image keeps its original license and is accompanied by a JSON sidecar with the following fields: image name; direct URL and source page; uploader, upload time, and original date; width and height; author; license, license URL, and license slug; usage terms; description; and categories. An attribution index generated from the sidecars satisfies the attribution and share-alike terms.

E.2 Statistics and Comparison with Real-ISR Benchmarks

We characterize LucidLR by its semantic content, resolution, and degradations, and compare it with three representative real-world low-quality benchmarks, RealLQ250, RealSR, and DRealSR, annotated with the same model and taxonomy. All statistics refer to the 20,000 released images.

Semantic content and degradations. Multimodal large language models have been used to identify degradations and plan restoration [9]; here, the same DeepSeek-V4.1-Flash call that screens each image also assigns one of 13 semantic categories, a single *primary* degradation, and the set of all visible degradation types (prompt at the end of this section). Fig. 10 compares the resulting distributions. LucidLR covers all 13 semantic categories without a dominant one (architecture 23.8%, people 22.1%, and no other category above 10%), whereas the benchmarks concentrate on a few scene types, e.g., objects, text, and architecture in RealSR. The degradation statistics show the same contrast: defocus blur is the primary degradation in 82.0% and 91.4% of the RealSR and DRealSR images

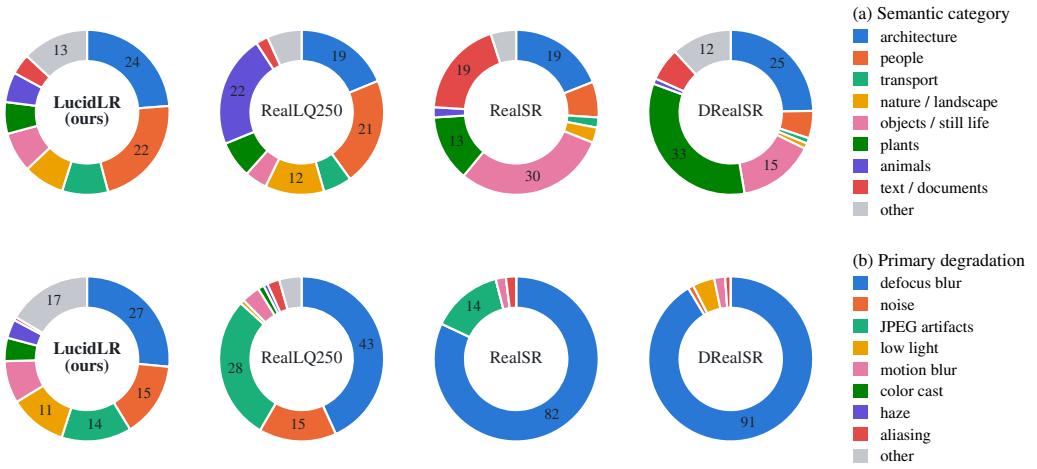

Figure 10: (a) Semantic category and (b) primary degradation of LucidLR and three real-world benchmarks, annotated with the same model and taxonomy. Each category has the same color and position in every chart; categories outside the eight shown are merged into *other*, and wedges of at least 10% give their percentage. The benchmarks are dominated by a few scene types and by defocus blur, whereas LucidLR spreads over many categories and degradations.

and in 43.2% of RealLQ250, but in only 26.5% of LucidLR, whose primary degradations spread over defocus blur, noise, JPEG artifacts, low light, motion blur, color cast, haze, and overexposure. Counting all visible degradations rather than only the primary one, LucidLR images frequently combine several of them, and types such as overexposure, lens flare, and occlusion, which occur in at most 2.0% of the images of any benchmark, appear in 3.0–9.8% of LucidLR. These annotations characterize the datasets and are not used for training; because all images are stored as JPEG, codec damage is almost always labeled `jpeg_artifacts`, so this label and `compression_artifacts` should be read together.

Resolution. LucidLR is not resized to a common size: every image keeps the resolution at which it was collected, so resolutions vary widely (Table 13). The median image is 1080×960 (1.05 MP) with a median short side of 853 pixels; 65.7% of the images have a short side of at least 512 pixels and 46.0% of at least 1024 pixels, while the long tail reaches 172 MP. 70.2% of the images are landscape, 28.4% portrait, and 1.3% square. This spread of resolutions complements the fixed-size crops of existing benchmarks.

Table 13: Short-side distribution of the 20,000 LucidLR images.

Short side (px)	<256	256–384	384–512	512–768	768–1024	1024–1536	1536–2048	≥ 2048
Images	2,068	1,720	3,064	2,528	1,416	2,752	2,597	3,855
Share	10.3%	8.6%	15.3%	12.6%	7.1%	13.8%	13.0%	19.3%

Annotation prompt. The first screening pass, the semantic category, and the degradation labels come from a single call with the following prompt, used verbatim for all four datasets.

Prompt E.1 (Screening and Annotation of LucidLR Images)

You are a dataset curator preparing a public image dataset of real-world degraded photographs. Analyze the single input image and return ONE JSON object.
Definitions of the degradation labels (use these exact strings):
- **motion_blur**: directional blur caused by camera or object motion
- **defocus_blur**: out-of-focus blur or shallow depth-of-field
- **noise**: visible sensor noise, grain, chroma noise

- **jpeg_artifacts**: JPEG blocking, mosquito noise, DCT artifacts
- **compression_artifacts**: generic codec artifacts beyond JPEG
- **aliasing**: jagged edges or resampling artifacts
- **haze / fog**: atmospheric scattering reducing contrast
- **rain_streaks / snow**: visible precipitation structures
- **low_light**: severe underexposure or dark scenes
- **overexposure**: clipped highlights or blown-out regions
- **color_cast**: global color tint due to white balance failure
- **color_distortion**: abnormal color reproduction
- **dirty_lens**: stains, droplets, or smears on the lens
- **occlusion**: partial obstruction by objects
- **lens_flare**: bright streaks or glare caused by strong light
- **other**: degradation exists but does not match the above labels
- **none**: image is mostly clean with no obvious degradation

Semantic categories (choose the category covering the most salient subject; if people occupy the main subject area, choose people):

- **people**: humans as main subject (portraits, crowds, ceremonies, workers)
- **architecture**: buildings, monuments, bridges, streets, cityscapes
- **nature_landscape**: mountains, forests, water, sky, fields, beaches
- **animals**: any non-human animal as main subject
- **plants**: flowers, leaves, trees as the principal subject (close-ups)
- **food**: dishes, ingredients, drinks
- **text_document**: signs, posters, book pages, maps, screenshots, forms where text dominates
- **art_illustration**: paintings, drawings, engravings, cartoons, logos, graphics, renders
- **objects_still_life**: equipment, products, tools, close-ups of inanimate objects
- **indoor_scene**: interiors, rooms, museums, stage settings
- **transport**: vehicles, trains, aircraft, ships, roads as main subject
- **events_sports**: sports, performances, concerts, military parades
- **other**: none of the above

Return exactly this JSON, no markdown, no commentary:

```
{
  "nsfw": <true if the image contains explicit nudity, sexual content, sex acts,
           graphic gore or graphic violence; suggestive but non-explicit content is NOT nsfw>,
  "nsfw_type": one of [none, nudity_sexual, suggestive, gore_violence, drugs,
               hate_symbols, other],
  "nsfw_confidence": <0.0-1.0>,
  "is_photograph": <true for camera/photographic images; false for drawings, paintings,
                    diagrams, maps, logos, screenshots, 3D renders>,
  "semantic_category": one of the semantic categories above,
  "primary_degradation": <one label from the degradation list, the dominant one>,
  "degradation_types": <1-6 labels from the degradation list, strongest first;
                        if the image is mostly clean use ["none"]>,
  "degradation_confidence": <0.0-1.0>
}
```

Be conservative: only list degradations clearly visible in the image.

F Broader Impacts

Potential benefits. Generative Real-ISR is increasingly used to enhance personal photographs, archival material, and images captured under poor conditions. By rewarding outputs that remain supported by the LR evidence, LucidNFT aims to reduce hallucinated text, faces, and structures that viewers may otherwise mistake for recovered detail, and LucidConsistency provides a reference-free signal for flagging outputs whose content departs from the input.

Potential risks. Improved faithfulness does not make generative SR outputs reliable evidence: the restored details remain samples from a learned prior, and both rewards can be imperfect (Sec. 6). Outputs of LucidNFT should therefore not be used for forensic analysis, identification of individuals, or medical decisions without independent verification. More realistic restoration may also make manipulated or fabricated imagery harder to distinguish from genuine photographs. LucidLR contains photographs of people collected from Wikimedia Commons; we screen it for explicit content, personal data, and hate symbols, release it only under licenses that permit redistribution, and record source

and license information for every image (Appendix E.1), but users remain responsible for complying with these licenses and with applicable personality rights.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes].

   Justification: The abstract and introduction state the main contributions, including Lucid-Consistency, decoupled reward normalization, and LucidLR. The experimental sections evaluate these claims through human-aligned faithfulness evaluation, rollout advantage analysis, quantitative comparisons, and ablations.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes].

   Justification: The conclusion contains a dedicated Limitations paragraph covering extreme degradations, reward-hacking risk, the domain coverage of LucidLR, the dependency of LucidConsistency on its Qwen3-VL-Embedding backbone, its dual role as reward and evaluation metric, and the absence of an external LR-faithfulness benchmark.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs.
   - The authors should reflect on the factors that influence the performance of the approach.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - The authors should use their best judgment and recognize that transparency about limitations helps preserve the integrity of the community.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A].

   Justification: The paper is primarily empirical and does not present formal theoretical results or proofs. The mathematical formulations describe the training objectives and reward normalization procedure.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.

- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material.
- Any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper?

   Answer: [Yes].

   Justification: The paper describes the model backbones, datasets, training settings, reward definitions, rollout settings, evaluation protocols, and baselines used in the experiments. Additional degradation ranges, dataset statistics, and implementation details are provided in the appendix or supplementary material.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Reproducibility can be provided via code, detailed instructions, hosted models, released checkpoints, or other appropriate means.
   - If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
   - If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
   - If the contribution is a new model, there should be a way to access or reproduce this model.
   - Authors should describe the particular avenue they provide for reproducibility when full reproduction is difficult.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes].

   Justification: LucidLR is released as a pre-fetched image bundle and as a metadata index with a reference downloader, each image with a JSON sidecar recording its source and license (Appendix E.1); implementation resources are released together with the dataset.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines for more details.
   - While release of code and data is encouraged, [No] is an acceptable answer when justified.
   - The instructions should contain the exact command and environment needed to reproduce the results.
   - The authors should provide instructions on data access and preparation.
   - The authors should provide scripts to reproduce all experimental results for the proposed method and baselines, or state which ones are omitted and why.
   - At submission time, authors should release anonymized versions if applicable.
   - Providing as much information as possible in supplemental material is recommended.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details necessary to understand the results?

   Answer: [Yes].

   Justification: The implementation details specify the backbones, LoRA settings, training resolutions, learning rates, batch sizes, rollout counts, inference steps, rewards, optimizer, and evaluation datasets. Additional controlled degradation settings are provided in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail necessary to appreciate the results.
   - Full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No].

   Justification: The paper reports extensive comparisons across multiple datasets, metrics, and ablations, but does not report error bars or formal statistical significance tests due to the high computational cost of repeated RL fine-tuning runs. Human-aligned evaluation is reported using aggregate agreement metrics.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if results are accompanied by error bars, confidence intervals, or statistical significance tests.
   - The factors of variability captured by the error bars should be clearly stated.
   - The method for calculating error bars should be explained.
   - The assumptions made should be given.
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it.
   - For asymmetric distributions, authors should avoid misleading symmetric error bars.
   - If error bars are reported, the text should explain how they were calculated and reference the corresponding figures or tables.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources needed to reproduce the experiments?

   Answer: [Yes].

   Justification: The paper reports the GPU type and number used for LucidConsistency and LucidNFT training, together with training steps, batch sizes, rollout counts, and training resolutions. These details provide the main compute requirements for reproducing the reported experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers, CPU or GPU, internal cluster or cloud provider, including relevant memory and storage.
   - The paper should provide the compute required for individual experimental runs and estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes].

   Justification: The work uses public or standard research datasets and includes filtering procedures for the newly collected real-world degraded images. The human annotation protocol is limited to image quality and faithfulness judgments and does not collect sensitive personal information.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity where required.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes].

    Justification: Appendix F discusses the potential benefit of more LR-faithful generative restoration and the risks that restored details are mistaken for reliable evidence, that more realistic restoration hampers the detection of manipulated imagery, and that LucidLR contains photographs of people, together with the screening and licensing measures we apply.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why.
    - Examples of negative societal impacts include malicious or unintended uses, fairness considerations, privacy considerations, and security considerations.
    - Foundational research not tied to deployment may still have direct paths to negative applications.
    - Authors should consider harms from intended use, incorrect results, and misuse.
    - If there are negative societal impacts, authors may discuss mitigation strategies.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse?

    Answer: [Yes].

    Justification: For LucidLR, every image is screened in two passes by a vision-language model (DeepSeek-V4.1-Flash), and images with explicit sexual content, photographic gore, personal data, or hate symbols are removed (Appendix E.1). Only images whose Commons licenses permit redistribution are included, and images with uncertain redistribution status are kept as URL-only entries.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models with high misuse or dual-use risk should be released with safeguards such as usage restrictions or safety filters.
    - Scraped datasets could pose safety risks, and authors should describe how unsafe images were avoided.
    - Providing effective safeguards is challenging, but authors are encouraged to make a best-faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets used in the paper properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes].

Justification: The paper cites the datasets, models, and methods used, including RealSR, DRealSR, RealLQ250, LSDIR, Real-ESRGAN, Qwen3-VL-Embedding, DiffusionNFT, and the evaluated SR baselines. LucidLR is collected from Wikimedia Commons; each image keeps its original license (mainly CC BY and CC BY-SA, plus public-domain and CC0 files), and author and license information is recorded per image for attribution.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license should be included for each asset.
- For scraped data from a particular source, the copyright and terms of service should be provided.
- If assets are released, license, copyright information, and terms of use should be provided.
- For repackaged datasets, both the original and derived licenses should be provided.
- If licensing information is unavailable online, authors are encouraged to contact asset creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes].

Justification: The paper introduces LucidLR and describes its source, scale, screening pipeline, licensing, release format, per-image metadata schema, and intended usage, together with semantic, resolution, and degradation statistics (Appendices E.1 and E.2).

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate dataset, code, and model details via structured templates.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, assets should be anonymized if applicable.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation?

Answer: [Yes].

Justification: The paper describes both human studies, the evaluator-alignment ranking study and the model-level pairwise preference study, including the LR-faithfulness instructions, the candidate presentation format, the number of annotators and judgments, and, for the model-level study, inter-annotator agreement (Fleiss' κ). Full annotation instructions and interface details are provided in the appendix.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether IRB approvals or equivalent approval/review were obtained?

    Answer: [N/A].

    Justification: The human study consists of expert judgments on image restoration faithfulness and does not collect personal, sensitive, or behavioral data from participants. The study follows the authors' institutional requirements for low-risk annotation studies.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval or equivalent may be required for any human subjects research.
    - Authors are expected to adhere to the NeurIPS Code of Ethics and their institutional guidelines.
    - For initial submissions, do not include information that would break anonymity.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research?

    Answer: [N/A].

    Justification: LLMs are not an important, original, or non-standard component of the proposed method. A vision-language model (DeepSeek-V4.1-Flash) is used only for dataset curation and characterization, namely content screening and semantic/degradation annotation of LucidLR, and this use is described in Sec. 4.3 and Appendices E.1–E.2.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to the NeurIPS LLM policy in the handbook for what should or should not be described.