CoRe: Recovery-Driven Head Routing for Real-Time Streaming Video Relighting

Anonymous authors
Paper under double-blind review

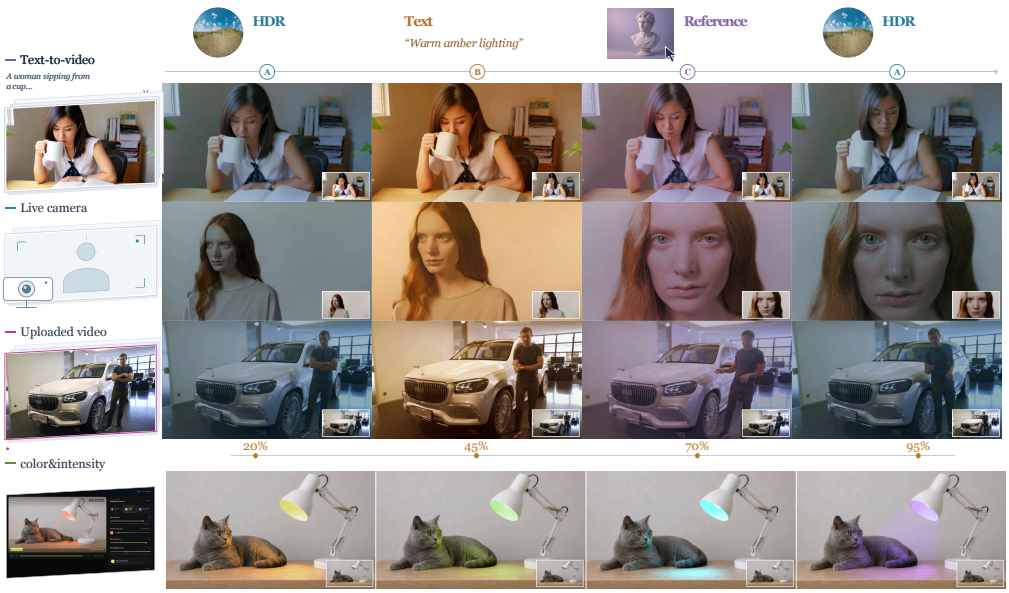

Figure 1: Editable streaming relighting with HDR, reference, and text controls.

Abstract

Interactive video relighting lets users change the illumination of a video while it plays. Streaming generators achieve the required speed and temporal coherence by reusing cached attention keys and values (K/V), but the cache also remembers the old light: after a lighting edit, reused states leak stale shadows and colors into new frames, whereas clearing or rebuilding the cache sacrifices continuity or latency. We observe that this conflict is concentrated in a subset of attention heads, and present **CoRe** (**Co**unterfactual **Re**covery), which routes each head either to persistent video memory or to the current light. Video-memory heads keep the rolling video cache, while light-control heads attend to K/V computed from the current lighting condition and never store them across blocks. CoRe learns how many light-control heads each layer receives, and which ones, through lighting-switch recovery: starting from a cache built under light A, the model must recover the output that the same video would receive under light B, subject to a global head budget. The learned routing is fixed before deployment, so an edit requires neither cache rebuilding nor online head selection. A single interface accepts HDR environment maps, reference images, and text. On minute-scale text-conditioned streams, CoRe reduces OldRes from 0.18 to 0.07 relative to uniform cache reuse at 23.8 FPS end-to-end on a single GPU, and it achieves the lowest LF among compared methods on short fixed-light clips for all three control modalities.

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

1 INTRODUCTION

Lighting shapes how a scene feels: the same shot can look warm, tense, or clinical depending only on how it is lit. Video relighting changes this illumination after capture while preserving scene content, geometry, and motion, with applications in film post-production, virtual production, and live video communication (Pandey et al., 2021; Ma et al., 2026). Its most natural form is interactive: a user adjusts the light, watches the result, and keeps editing while the video plays (Fig. 1). This setting demands three properties at once: precise control over light direction, intensity, and color; immediate feedback after each edit; and content preservation and temporal continuity over long videos, including across repeated lighting changes.

Diffusion models have made relighting remarkably capable. Conditioned on HDR environment maps, reference images, or text prompts, they synthesize shadows, highlights, and color shifts that follow the target light while preserving source content (Zhang et al., 2025; Jin et al., 2024; Fang et al., 2025; Zhou et al., 2025), and recent work extends them to dynamic lighting and streaming inference (Xiao et al., 2026; Ma et al., 2026). To generate long videos efficiently, streaming models reuse key–value (K/V) caches or feature banks, which avoid recomputation and carry identity, motion, and scene context from one block to the next (Yang et al., 2026a; Liang et al., 2025a). This memory, however, is built for a fixed condition: it assumes that states computed under the previous light remain valid for the next block, and a lighting edit breaks exactly this assumption. The result is a dilemma between memory and control. Cached states entangle scene context that should persist with appearance shaped by the light that has just been replaced. Keeping the cache carries stale colors and shadows into subsequent frames; clearing it discards the context that keeps identity and motion coherent; and rebuilding it under the new condition (Yang et al., 2026a) adds computation at every edit. Interactive relighting must retain scene memory but forget old light without rebuilding history.

Our key observation is that this entanglement is not spread evenly across attention heads. When the lighting of a fixed video is swapped, a minority of heads accounts for a disproportionate share of the K/V change, while the rest stay close to a low baseline. In a patching probe (Fig. 13), injecting old-light K/V into the most light-sensitive heads increases stale-light leakage more than injecting it into the least sensitive ones, while swapping in another clip's K/V degrades geometry more in the least sensitive heads. The question is therefore not *whether* to keep the cache, but *which* heads should remember the video and which should look at the current light. Sensitivity alone cannot answer it: the size of a head's K/V change does not reveal whether redirecting that head improves the relit output. We answer it with **CoRe**, which learns a fixed partition between *video-memory heads* that retain temporal context and *light-control heads* that use video-derived queries to read only current-light K/V, without caching them across blocks. Lighting-switch recovery trains this routing under a global head budget, so edits refresh the light-control branch without rebuilding video history or selecting heads online. We evaluate CoRe on fixed-light clips, paired multi-illumination scenes (Murmann et al., 2019), and minute-scale streams with lighting edits. On the latter, CoRe reduces old-light residuals and edit discontinuities relative to uniform cache reuse and LongLive recaching, and runs faster than both at 23.8 FPS end-to-end on one GPU.

Our contributions are threefold. **(1) Problem and insight.** We cast lighting edits in streaming relighting as a head-level memory-routing problem and show that stale-light effects concentrate in a subset of attention heads. **(2) Method.** We propose CoRe, which learns a fixed, budgeted partition of heads into persistent video memory and non-persistent current-light control through lighting-switch recovery, jointly learning per-layer counts and head identities. **(3) Evaluation.** We introduce a switch-based protocol that measures old-light residuals and target-reaching time together with relighting fidelity, temporal continuity, and inference cost, and show that CoRe improves edit response at 23.8 FPS end-to-end throughput.

2 RELATED WORK

Image and video relighting. Learned relighting has long relied on light-stage and multi-illumination data, from portrait relighting and background replacement (Pandey et al., 2021) and real-time video portrait relighting (Zhang et al., 2021) to multi-illumination captures of real scenes (Murmann et al., 2019). Diffusion models have since extended relighting to in-the-wild

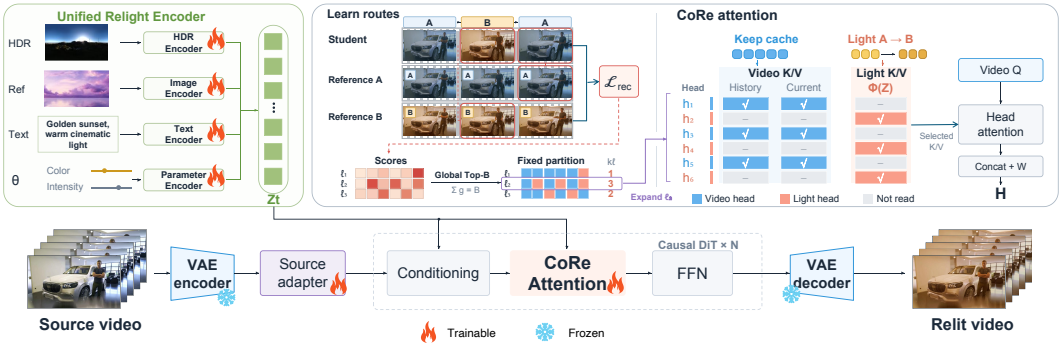

Figure 2: **Overview of CoRe.** Lighting-switch recovery learns a fixed head partition under a global budget. Video-memory heads retain rolling video K/V, while light-control heads read only current-light K/V without persisting them, enabling lighting edits without cache rebuilding.

content. IC-Light scales illumination editing by imposing consistent light transport during training (Zhang et al., 2025), and Neural Gaffer relights objects under target environment maps (Jin et al., 2024). For video, RelightVid learns temporally consistent relighting from background-video, text, or environment-map conditions (Fang et al., 2025); Light-A-Video is training-free, combining cross-frame light attention with progressive light fusion (Zhou et al., 2025); and TC-Light refines relit long, dynamic videos through two-stage post-optimization (Liu et al., 2025b). DiffusionRenderer, UniRelight, and UniLumos bring intrinsic or geometric cues into video relighting (Liang et al., 2025b; He et al., 2025; Liu et al., 2025a). Relit-LiVE jointly predicts relit videos and viewpoint-aligned environment maps (Xiao et al., 2026), LightCtrl follows user-specified light trajectories (Peng et al., 2026), and LiveLight, closest to our setting, performs real-time streaming relighting with interactive 3D lighting control (Ma et al., 2026). These methods determine how the relit output should look; we study what a streaming model should remember when the requested light changes mid-stream.

Streaming video generation and editing. Diffusion Forcing uses independent per-token noise levels for causal generation (Chen et al., 2024), CausVid distills a bidirectional model into a few-step causal generator (Yin et al., 2025), and Self Forcing trains on self-generated histories and extrapolates with a rolling KV cache (Huang et al., 2025). StreamV2V reuses historical K/V (Liang et al., 2025a), LiveEdit combines causal distillation with a mask cache (Wang et al., 2026), and SANA-Streaming uses a hybrid diffusion transformer with cycle-reverse regularization (Zhao et al., 2026). For prompt changes, LongLive rebuilds cached states through KV-recache (Yang et al., 2026a), Anchor Forcing re-caches from anchor memories (Yang et al., 2026b), and Echo-Forcing maintains hierarchical scene memory under a fixed KV budget (Wu et al., 2026). We build on LongLive's causal backbone, rolling window, and frame sink, and compare its recaching directly with CoRe's fixed routing between video history and current-light K/V, which requires no cache rebuilding.

KV cache management and head specialization. Long-context LLM inference reduces KV storage by keeping selected tokens: attention sinks with a recent window (Xiao et al., 2024), heavy-hitter tokens (Zhang et al., 2023), prompt positions chosen from an observation window (Li et al., 2024), or entries scored by context reconstruction (Kim et al., 2025). Another line of work exploits the fact that attention heads behave differently. Ada-KV allocates eviction budgets across heads using an upper bound on the attention-output loss (Feng et al., 2025); DuoAttention identifies retrieval heads that keep the full cache and streaming heads that keep only sinks and recent tokens (Xiao et al., 2025); HACK compresses contextual and structural heads differently in visual autoregressive models (Qin et al., 2026); and MoH lets each token select and weight a subset of heads (Jin et al., 2025). In video diffusion transformers, Sparse-vDiT finds attention sparsity patterns that are largely input-independent and tied to layer depth and head position (Chen et al., 2026). These methods decide how much history each head keeps, or which heads each token uses. Lighting edits raise a different question: *which source* each head should read, persistent video history or the current light. CoRe learns this assignment from its effect on the relit output after a lighting switch.

3 METHOD

CoRe performs streaming video relighting under time-varying illumination. Given a source video and a sequence of target lighting conditions, the model generates relit video blocks autoregressively. The lighting can be specified in two modes: a *multimodal-input mode*, which switches among HDR, text, and reference-image conditions, and a *parameter-control mode*, which sets the lighting directly through continuous parameters. We build on LongLive (Yang et al., 2026a), adding a source-video adapter and light conditioning (Sec. 3.1). To accommodate lighting edits while preserving temporal context, CoRe routes each attention head either to the video history or to the current light (Sec. 3.2). The number of light-routed heads per layer and their identities are learned under a global budget through lighting-switch recovery (Sec. 3.3) and then fixed for inference (Sec. 3.4).

3.1 PRELIMINARIES

Streaming relighting backbone. Let $X_{1:T}$ be the source-video blocks and c_t the target light at block t. A frozen VAE encodes the video, and a source adapter injects its appearance and motion features into the causal generator. For $q \in \{\text{HDR}, \text{reference}, \text{text}, \text{param}\}$, light tokens are

$$Z_t = P_q\big(E_q(c_t)\big) \in \mathbb{R}^{M \times d_z}, \tag{1}$$

where E_q encodes the condition and P_q projects its output to M tokens of width d_z. Parameter control writes $c_t = \theta_t$ (current color and intensity) as a text instruction, using the text encoder E_{param} and projection P_{param}; E_{param} is trained on text pairs including these continuous parameter instructions (Sec. A.1). A light-conditioning module feeds Z_t into the backbone's conditional-attention pathway. Generation retains the backbone's causal sampling schedule, rolling window, and frame sink; video attention reads the current block and cached K/V of previously generated blocks. For the model being trained (*student*), $\widehat{z}_t$ denotes the sampled clean latent of block t with D_t elements. Cached history is detached, and $\mathrm{sg}(\cdot)$ denotes stop-gradient.

Relighting pretraining. We pretrain on paired videos $(X_{1:T}, Y^c_{1:T}, c)$ sharing content and motion under a constant target light c, using self-generated history in the video-attention cache:

$$\mathcal{L}_{\text{rel}} = \lambda_{\text{FM}}\mathcal{L}_{\text{FM}} + \lambda_{\text{rgb}}\mathcal{L}_{\text{rgb}} + \lambda_{\text{perc}}\mathcal{L}_{\text{perc}} + \lambda_{\text{temp}}\mathcal{L}_{\text{temp}}, \tag{2}$$

where $\mathcal{L}_{\text{FM}}$ applies flow matching to latent velocity prediction (Lipman et al., 2023); $\mathcal{L}_{\text{rgb}}$ and $\mathcal{L}_{\text{perc}}$ apply ℓ_1 RGB and LPIPS (Zhang et al., 2018) to decoded frames; and $\mathcal{L}_{\text{temp}}$ matches frame-to-frame changes between generated and target videos, excluding scene cuts. The resulting F_0 reuses the video cache in all heads and initializes CoRe; a frozen copy supplies references for lighting-switch supervision (Sec. 3.3). $\mathcal{L}_{\text{rel}}$ remains active during routing training; the source adapter and light-conditioning pathway are used throughout training and inference.

3.2 PER-HEAD VIDEO AND LIGHT ROUTING

We add routing to a set $\mathcal{R}$ of video self-attention layers. Layer $\ell \in \mathcal{R}$ has H_ℓ heads of dimension d_ℓ, and each head h has two candidate routes: a video route and a current-light route. For queries Q, keys K, values V, and an additive mask $\mathcal{M}$, we write

$$\mathrm{Attn}_\ell(Q, K, V; \mathcal{M}) = \mathrm{softmax}\left(\frac{QK^\top}{\sqrt{d_\ell}} + \mathcal{M}\right) V. \tag{3}$$

Video route. The current video states are projected to queries, keys, and values; the video query $Q^V_{\ell h,t}$ and key $K^V_{\ell h,t}$ follow the backbone's normalization and rotary positional encoding (RoPE). Given the cached history $(C^K_{\ell h,t-1}, C^V_{\ell h,t-1})$, which contains the frame sink and the rolling window, the video-route output is

$$O^V_{\ell h,t} = \mathrm{Attn}_\ell\big(Q^V_{\ell h,t}, [C^K_{\ell h,t-1}; K^V_{\ell h,t}], [C^V_{\ell h,t-1}; V^V_{\ell h,t}]; \mathcal{M}^V\big)\,, \tag{4}$$

where $[\,\cdot\,;\cdot\,]$ concatenates along the token axis and $\mathcal{M}^V$ is the causal mask.

Current-light route. The light route reuses the head's query projection and takes the normalized query before RoPE, denoted $Q^L_{\ell h,t}$. Its keys and values are projected from the current light tokens by a per-head projector $\Phi_{\ell h}$:

$$(K^L_{\ell h,t}, V^L_{\ell h,t}) = \Phi_{\ell h}(Z_t), \qquad O^L_{\ell h,t} = \mathrm{Attn}_\ell\big(Q^L_{\ell h,t}, K^L_{\ell h,t}, V^L_{\ell h,t}; \mathbf{0}\big)\,. \tag{5}$$

We denote the parameters of all projectors by ψ. Light keys follow the backbone's head normalization but receive no RoPE: light tokens have no frame index, and assigning them video positions would impose a temporal order they do not have. The zero mask makes all M light tokens visible to every query. Light K/V are recomputed from Z_t and never stored in the rolling video cache.

Route design. We call heads on the light route *light-control heads* and heads on the video route *video-memory heads*. A light-control head performs video-to-light attention: its queries remain video-derived, but its keys and values come only from the current light tokens. It therefore attends neither to the cached history nor to the current block's video tokens, which removes its direct access to historical video K/V and skips its video-window attention. Information exchange among video tokens is left to the video-memory heads, while the residual stream preserves each token's own features. As alternatives, we consider *augmentation*, which concatenates light K/V with video K/V so that a head still reads the rolling cache: Concat-all applies it to every head, and Augment-mask applies it only to CoRe's selected heads. We use replacement; Sec. 4.3 compares all three.

Jointly learning layer counts and head identities. Let $\mathcal{I} = \{(\ell, h) : \ell \in \mathcal{R},\ 1 \le h \le H_\ell\}$ index the $H_{\text{tot}} = \sum_{\ell \in \mathcal{R}} H_\ell$ candidate heads, and write $g_{\ell h}$ for g_i with $i = (\ell, h)$. We fix a global budget B, with $1 \le B \le H_{\text{tot}} - 1$, on the number of light-control heads. Each head $i \in \mathcal{I}$ has a learnable score a_i shared across videos, blocks, and conditions. Following the soft Top-k operator of Gupta et al. (2024), the budget-constrained gates are

$$g_i = \sigma\left(\frac{a_i - \nu}{\tau}\right), \qquad \sum_{i \in \mathcal{I}} g_i = B, \tag{6}$$

where σ is the logistic sigmoid, $\tau > 0$ is a temperature, and the threshold ν is solved for the current scores and temperature. The sum decreases continuously and strictly in ν from H_{tot} to 0, so ν exists and is unique for every admissible B. Each routed layer mixes its two routes head by head:

$$O_{\ell,t} = \text{Concat}_{h=1}^{H_\ell}\left[(1 - g_{\ell h})\, O^V_{\ell h,t} + g_{\ell h}\, O^L_{\ell h,t}\right] W^O_\ell, \tag{7}$$

where W^O_ℓ is the layer's output projection. The soft count of light-control heads in layer ℓ is $\tilde{k}_\ell = \sum_h g_{\ell h}$; because all layers share the threshold ν, they compete for the same budget. We call this soft-gated stage *routing search*. During search, both routes are evaluated for every candidate head, and video K/V are formed from the student's own generated states.

Writing $w_i = g_i(1 - g_i)$ and differentiating the budget constraint implicitly gives

$$\frac{\partial g_i}{\partial a_j} = \frac{w_i}{\tau}\left(\mathbf{1}[i{=}j] - \frac{w_j}{\sum_{r \in \mathcal{I}} w_r}\right). \tag{8}$$

Summing over i gives zero for every j, so each gradient step redistributes the budget among heads without changing its total. The scores and trainable model weights are optimized jointly (Sec. 3.4): the budget fixes the total number of light-control heads, and the training signal determines how they are distributed across layers and heads.

3.3 Recovery Under Lighting Changes

Reference trajectories. For a light c^u, let $S^u_{1:T}$ denote the clean latent targets for the source video under that light. When a target video aligned with the source in content and time is available under c^u, $S^u_{1:T}$ is its VAE encoding; otherwise, it is the output of the frozen F_0 run under the constant light c^u. This choice is fixed for each source–light pair, and the targets use the student's latent scaling and block indexing. Reference generation uses the same source video, sampling schedule, and noise as the student, while each model maintains its own video history.

Switched rollouts. We sample two lights c^{A} and c^{B} and two edit positions $2 \le t_1 < t_2 \le T$. The student starts under c^{A}, switches to c^{B} at block t_1, and returns to c^{A} at block t_2, generating continuously and carrying its video history through both edits. Let $u_t \in \{\text{A}, \text{B}\}$ be the active light: block t uses $c_t = c^{u_t}$ and target $S^{u_t}_t$. For a positive integer K_{rec}, the recovery windows

$$\mathcal{W} = \bigcup_{s \in \{t_1, t_2\}} \left\{t :\ s \le t \le \min(s + K_{\text{rec}} - 1,\, T)\right\}$$

collect the first K_{rec} blocks after each edit, with overlapping blocks counted once. The recovery loss supervises the student's clean latents in these windows:

$$\mathcal{L}_{\mathrm{rec}} = \frac{1}{|\mathcal{W}|} \sum_{t \in \mathcal{W}} \frac{\|\widehat{z}_t - \mathrm{sg}(S_t^{u_t})\|_F^2}{D_t}. \tag{9}$$

It asks the student to match the constant-light reference for the active light, even though its cached history was generated under the previous light. Post-edit blocks outside $\mathcal{W}$ are supervised with the RGB and LPIPS losses against the decoded targets, which gives $\mathcal{L}_{\mathrm{sw}}$.

3.4 Optimization and Streaming Inference

Routing search. Starting from F_0, we jointly optimize the head scores and the trainable model parameters under the budget constraint in Eq. (6) with

$$\mathcal{L}_{\mathrm{search}} = \mathcal{L}_{\mathrm{rel}} + \lambda_{\mathrm{sw}} \mathcal{L}_{\mathrm{sw}} + \lambda_{\mathrm{rec}} \mathcal{L}_{\mathrm{rec}}. \tag{10}$$

$\mathcal{L}_{\mathrm{rel}}$ preserves the relighting ability of F_0, while $\mathcal{L}_{\mathrm{sw}}$ and $\mathcal{L}_{\mathrm{rec}}$ teach the model to follow lighting edits. Rollouts use generated history with gradients stopped at block boundaries. Although the cached K/V are detached, the gates still receive gradients through the current block, because they determine whether each head reads the cache or the current light. The temperature τ is annealed during search so that the gates become increasingly sharp.

Fixed head assignment. After search, the B heads with the highest scores across all routed layers become light-control heads and the remaining heads become video-memory heads. Because every gate is monotonic in its score, this is the hard Top-B counterpart of the soft gates. Let k_ℓ denote the resulting number of light-control heads in layer ℓ, so that $\sum_{\ell \in \mathcal{R}} k_\ell = B$. We then freeze the assignment, rebuild the training histories with fixed routes, then train the model parameters with the same objective to adapt the generator to one route per head. Section 4.3 analyzes this allocation.

Streaming inference. The head assignment is fixed across videos and lighting requests, and each routed head computes only its assigned route. In multimodal-input mode, an edit replaces the HDR, text, or reference-image condition; in parameter-control mode, the continuous parameters θ_t are written into the text instruction, and the new values take effect at block boundaries. In both modes, the edit updates Z_t, the base conditioning inputs, and the current-light K/V, while video-memory heads keep their rolling history and frame sink. Light K/V are never appended to the video history, and the source adapter continues to process incoming source blocks. Edits require neither cache rebuilding nor online head selection. Peak memory for each budget is reported in Table 10.

4 Experiments

4.1 Setup

Data and evaluation. We split $1{,}000$ source videos by identity into 800 training and 200 held-out videos. We synthesize relighting pairs from the training videos using Wan2.1-VACE (Jiang et al., 2025) and Editto (Bai et al., 2026), followed by filtering and refinement (Appendix A.1). Evaluation uses short clips (5–10 s), medium clips (10–30 s), and interactive streams (45.6–60 s) with one prescribed mid-stream edit, drawn from the held-out videos. MIT Multi-Illumination (Murmann et al., 2019) provides paired test scenes excluded from training. Methods use identical clips and conditions at 832×480 with 30 FPS sources; we average scores per video, then across videos.

Implementation and baselines. CoRe uses 29 routed layers with 12 heads each, a light-head budget $B{=}87$, and three latent frames per block. Training uses four H200 GPUs; runtime uses one H200 in BF16. LongLive-based models share a 1280×720 display upsampler. LongLive+Adapter denotes F_0; LongLive+Recache applies the official KV-recache operator to F_0 at the edit. External baselines use official checkpoints and are timed at 832×480. FPS denotes generator-only or end-to-end throughput, as specified in each caption.

Metrics. LF measures scale-normalized lighting-appearance error and is compared only within a modality. DINO measures source preservation; tOF and Flicker measure source-flow-aligned RGB error and luminance fluctuations, respectively. Jump measures switch-time luminance-change error

	Short (5–10 s)				Medium (10–30 s)			
	Relighting		Temporal		Relighting		Temporal	
Method	LF↓	DINO↑	tOF↓	Flk.↓	LF↓	DINO↑	tOF↓	Flk.↓
Text conditioned								
TC-Light (2025b)	5.893	0.731	0.0150	2.552	**5.861**	0.770	0.0207	2.828
Light-A-Video (2025)	6.128	0.636	0.0194	4.148	6.081	0.651	**0.0203**	4.360
LiveEdit (2026)	7.529	0.698	0.0188	3.302	7.690	0.658	0.0208	3.525
StreamV2V (2025a)	7.037	0.740	0.0203	3.717	6.761	0.759	0.0232	4.026
IC-Light (2025)	5.834	0.507	0.0549	14.574	6.078	0.504	0.0578	15.023
LongLive + Adapter (2026a)	6.082	**0.796**	0.0146	2.612	6.118	**0.809**	0.0226	2.891
CoRe-Text	**5.788**	0.782	**0.0142**	**2.498**	6.001	0.778	0.0214	**2.812**
HDR conditioned								
RelightVid (2025)	8.460	0.465	0.0197	3.130	–	–	–	–
Relit-LiVE (2026)	5.910	0.776	**0.0130**	**2.419**	–	–	–	–
Diff. Renderer (2025b)	6.880	0.826	0.0202	3.674	–	–	–	–
UniRelight (2025)	6.080	**0.874**	0.0148	2.614	–	–	–	–
UniLumos (2025a)	8.210	0.785	0.0280	5.027	8.180	0.749	0.0299	5.437
Neural Gaffer (2024)	6.640	0.800	0.0420	7.641	6.710	0.797	0.0492	7.918
LongLive + Adapter (2026a)	6.108	0.812	0.0148	2.624	6.152	**0.821**	0.0228	2.918
CoRe-HDR	**5.590**	0.804	0.0144	2.536	**5.730**	0.798	**0.0216**	**2.842**
Reference conditioned								
Relit-LiVE (2026)	6.210	**0.938**	0.0216	3.380	–	–	–	–
LongLive + Adapter (2026a)	6.068	0.918	0.0144	2.608	**6.168**	**0.902**	0.0230	2.926
CoRe-Ref	**5.859**	0.908	**0.0140**	**2.482**	6.303	0.896	**0.0212**	**2.798**

Table 1: Fixed-light relighting on short and medium videos. **Bold**/underline: best/second best within each modality and duration. LongLive+Adapter denotes F_0.

	Relighting		Temporal		Edit response		Streaming	
Method	LF↓	DINO↑	tOF↓	Flk.↓	OldRes↓	Jump↓	Lat.↓	FPS↑
LiveEdit (2026)	6.330	0.520	0.0290	5.490	0.48	18.40	0.352	2.84
StreamV2V (2025a)	6.420	0.720	0.0220	4.980	0.35	9.15	0.987	1.01
LongLive + Adapter (2026a)	6.217	0.822	0.0216	4.521	0.18	7.85	0.050	20.0
LongLive + Recache	6.184	**0.841**	0.0231	4.708	0.15	6.72	0.050	20.0
CoRe	**6.094**	0.802	**0.0210**	**4.385**	**0.07**	**5.57**	**0.042**	**23.8**

Table 2: Interactive streams with one mid-stream lighting edit (text-conditioned streams, 45.6–60 s). Jump is the extra luminance change beyond the correct switch; Flicker is the aligned high-frequency residual. Lat.: per-frame wall-clock time (s); FPS: its reciprocal, end to end for LongLive-based rows (Sec. 4.1). LongLive+Adapter denotes F_0.

against each method's constant-light references. T_{90} uses these references to measure the video time needed to reach 90% of the new-light luminance change and sustain it for nine frames. OldRes compares relative distances to old/new condition anchors over the first second after an edit; SwID measures identity consistency across the edit. Metric definitions are in Appendix A.2.

4.2 Quantitative and Qualitative Comparison

Fixed-light relighting. In Table 1, CoRe achieves the lowest short-clip LF across modalities and the best short-clip tOF and Flicker under text and reference control. For HDR, it also leads LF and temporal metrics on medium clips; on short clips, Relit-LiVE is slightly more stable and UniRelight has higher DINO. Relit-LiVE also has higher DINO on short reference clips. On medium text clips, TC-Light has lower LF and Light-A-Video lower tOF, while CoRe leads Flicker. On medium reference clips, CoRe improves tOF and Flicker over F_0 at a small LF cost. On MIT (Table 4), CoRe-HDR ranks first on all four metrics, followed by Relit-LiVE.



Figure 3: HDR-conditioned qualitative comparison. CoRe follows the target environment map while preserving subject structure across frames.



Figure 4: Reference-image comparison against commercial video generators. Columns span Frame 1 to the shortest rollout's last common frame.

Interactive streams with lighting edits. On interactive streams (Table 2), CoRe achieves the best LF, temporal stability, OldRes, Jump, and end-to-end throughput, though its DINO is below both LongLive baselines. LongLive recaching only partly reduces the stale-light residual of uniform cache reuse (F_0) and adds computation at each edit. CoRe's light-control heads skip video-window attention, giving it a throughput advantage. LiveEdit and StreamV2V retain larger old-light residuals and run far below real time.

Qualitative comparison. Figures 3, 8, and 9 compare methods on short videos under HDR, text, and reference control, respectively. CoRe retains scene structure while shading and color follow the requested condition. Figure 4 further compares reference-image relighting with commercial video generators on the same clip.

Human perceptual evaluation. In a pairwise user study with 50 participants, each person compares CoRe with each baseline on visual realism (VR), physical consistency (PC), and target-light alignment (LA) under anonymized and randomized presentation (Table 6). On short, medium, and long videos, participants prefer CoRe over every baseline on all three criteria. On the MIT stills the preferences are closer, and Relit-LiVE and UniLumos are slightly preferred for visual realism. Per-baseline results are in Appendix A.6.

4.3 Ablation Studies

Head allocation. A lighting-swap probe on F_0 shows that heads respond unevenly to a lighting change, and causal patching (Fig. 13) links high-sensitivity heads to stale-light leakage and lower-sensitivity heads to continuity. Table 9 examines how CoRe exploits this structure. The first three rows assign the same number of light-control heads to every layer and change only how the heads are chosen: at random, by a sensitivity probe, or by learning. Learned identities leave less old light

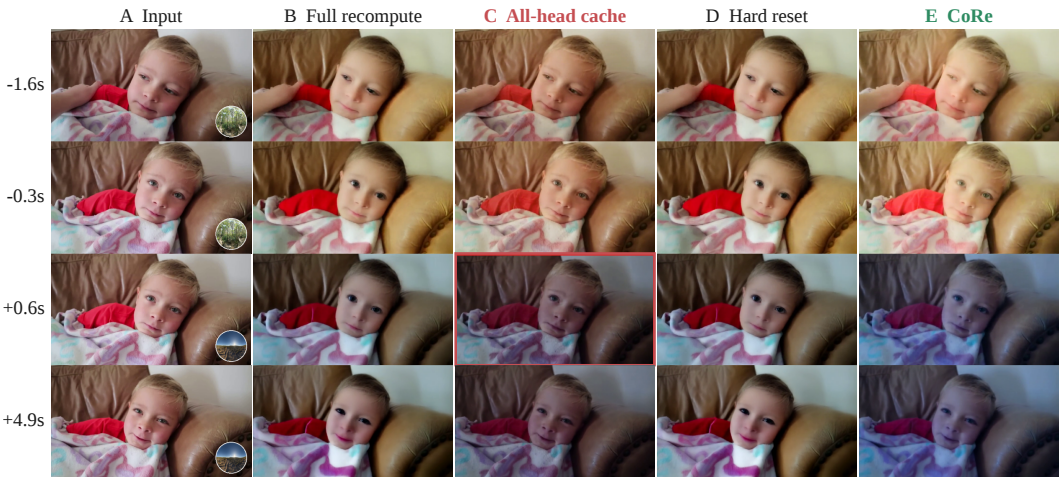

Figure 5: Cache-policy ablation. CoRe reduces stale-light leakage while retaining video memory.

and reach the new light faster than random or probe-selected heads, and learning the per-layer counts as well improves both further. The learned counts vary noticeably across layers.

Route design and budget. Concat-all and Augment-mask still let the light heads read the historical K/V, so a noticeably larger old-light residual remains and throughput stays close to that of F_0. Replacement removes this direct access to history and is clearly faster because selected heads skip video-window attention. Table 10 repeats CoRe at three budgets against a control that assigns the same number of light-control heads to every layer. Learning the counts lowers OldRes at every tested budget. A larger budget reduces cache memory and increases throughput but slightly lowers identity continuity; B=87 gives the lowest OldRes and is our default (Appendix A.7).

Cache policy. Table 3 and Fig. 5 compare cache handling at a lighting edit. All rows share the frame sink and rolling window. All-head cache reuses every cached head; hard reset discards the cache; full recompute rebuilds the history under the new light; CoRe refreshes the current-light K/V of its light-control heads and keeps the video cache. All-head cache carries the old light into the new segment, giving the highest OldRes, the longest T_{90}, and the largest jump at the edit. Hard reset removes most of the old light but breaks temporal continuity, so its Jump stays high and its T_{90} is longer. Full recompute gives the lowest OldRes and T_{90} but re-encodes the whole history at each edit. CoRe nearly matches it without rebuilding and gives the lowest Jump, as the retained video cache keeps the transition continuous. Figure 5 visualizes the two failure modes that CoRe mitigates: stale-light leakage under uniform reuse and loss of continuity under hard invalidation.

Policy	Jump↓	OldRes↓	T_{90}↓	LF↓
Full recompute	6.57	**0.04**	**0.70**	6.178
All-head cache	8.12	0.19	2.52	6.248
Hard reset	8.01	0.06	1.18	6.235
CoRe	**4.28**	<u>0.05</u>	<u>0.72</u>	**5.826**

Table 3: Cache-policy ablation on HDR streams. **Bold**/<u>underline</u>: best/second best in column.

5 Conclusion

We present CoRe, a recovery-trained head-routing method for streaming video relighting with mid-stream lighting edits. CoRe assigns attention heads to persistent video memory or non-persistent current-light control under a global head budget. The assignment is learned through lighting-switch recovery and fixed before deployment, allowing lighting edits without rebuilding the video cache or selecting heads online. A unified light-token interface supports HDR, reference-image, and text conditions. Experiments on fixed-light clips and interactive streams show improved lighting recovery while preserving source content and temporal continuity. Under the reported setup, CoRe achieves 23.8 FPS end-to-end throughput on a single H200 GPU.

REFERENCES

Qingyan Bai, Qiuyu Wang, Hao Ouyang, Yue Yu, Hanlin Wang, Wen Wang, Ka Leong Cheng, Shuailei Ma, Yanhong Zeng, Zichen Liu, Yinghao Xu, Yujun Shen, and Qifeng Chen. Scaling instruction-based video editing with a high-quality synthetic dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 37971–37981, 2026.

Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-0759.

Pengtao Chen, Xianfang Zeng, Maosen Zhao, Mingzhu Shen, Wei Cheng, Gang Yu, and Tao Chen. Sparse-vDiT: Unleashing the power of sparse attention to accelerate video diffusion transformers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 2957–2965, 2026. doi: 10.1609/aaai.v40i4.37287.

Ye Fang, Zeyi Sun, Shangzhan Zhang, Tong Wu, Yinghao Xu, Pan Zhang, Jiaqi Wang, Gordon Wetzstein, and Dahua Lin. Relightvid: Temporal-consistent diffusion model for video relighting. *arXiv preprint arXiv:2501.16330*, 2025.

Yuan Feng, Junlin Lv, Yukun Cao, Xike Xie, and S. Kevin Zhou. Ada-KV: Optimizing KV cache eviction by adaptive budget allocation for efficient LLM inference. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/a40ff56daab9f4808b1e18350c8a11ce-Abstract-Conference.html.

Nilesh Gupta, Fnu Devvrit, Ankit Singh Rawat, Srinadh Bhojanapalli, Prateek Jain, and Inderjit S. Dhillon. Dual-encoders for extreme multi-label classification. In *The Twelfth International Conference on Learning Representations*, 2024.

Kai He, Ruofan Liang, Jacob Munkberg, Jon Hasselgren, Nandita Vijaykumar, Alexander Keller, Sanja Fidler, Igor Gilitschenski, Zan Gojcic, and Zian Wang. Unirelight: Learning joint decomposition and synthesis for video relighting. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. In *Proceedings of ICCV*, pp. 17191–17202, 2025.

Haian Jin, Yuan Li, Fujun Luan, Yuanbo Xiangli, Sai Bi, Kai Zhang, Zexiang Xu, Jin Sun, and Noah Snavely. Neural gaffer: Relighting any object via diffusion. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-4481.

Peng Jin, Bo Zhu, Li Yuan, and Shuicheng Yan. MoH: Multi-head attention as mixture-of-head attention. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 28233–28255. PMLR, 2025. URL https://proceedings.mlr.press/v267/jin25l.html.

Jang-Hyun Kim, Jinuk Kim, Sangwoo Kwon, Jae W. Lee, Sangdoo Yun, and Hyun Oh Song. KVzip: Query-agnostic KV cache compression with context reconstruction. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/f4eaa4b8f2d08edb3f0af990d56134ea-Abstract-Conference.html.

Yuhong Li, Yingbing Huang, Bowen Yang, Bharat Venkitesh, Acyr Locatelli, Hanchen Ye, Tianle Cai, Patrick Lewis, and Deming Chen. SnapKV: LLM knows what you are looking for before generation. In *Advances in Neural Information Processing Systems*, volume 37, pp. 22947–22970, 2024. doi: 10.52202/079017-0722.

Feng Liang, Akio Kodaira, Chenfeng Xu, Masayoshi Tomizuka, Kurt Keutzer, and Diana Marculescu. Looking backward: Streaming video-to-video translation with feature banks. In *Proceedings of the International Conference on Learning Representations*, 2025a.

Ruofan Liang, Zan Gojcic, Huan Ling, Jacob Munkberg, Jon Hasselgren, Chih-Hao Lin, Jun Gao, Alexander Keller, Nandita Vijaykumar, Sanja Fidler, and Zian Wang. Diffusion renderer: Neural inverse and forward rendering with video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26069–26080, 2025b.

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

Pengwei Liu, Hangjie Yuan, Bo Dong, Jiazheng Xing, Jinwang Wang, Rui Zhao, Weihua Chen, and Fan Wang. Unilumos: Fast and unified image and video relighting with physics-plausible feedback. In *Advances in Neural Information Processing Systems*, volume 38, 2025a.

Yang Liu, Chuanchen Luo, Zimo Tang, Yingyan Li, Yuran Yang, Yuanyong Ning, Lue Fan, Junran Peng, and Zhaoxiang Zhang. Tc-light: Temporally coherent generative rendering for realistic world transfer. In *Advances in Neural Information Processing Systems*, volume 38, 2025b.

Yue Ma, Jiangming Wang, Yucheng Wang, Xilai Wang, Zhiyuan Li, Xinyu Wang, Hongyu Liu, Ruofan Liang, Songchun Zhang, Yuxuan Xue, and Qifeng Chen. LiveLight: Real-time streaming video relighting with interactive control. *ACM Transactions on Graphics*, 2026. URL `https://arxiv.org/abs/2608.01771`.

Lukas Murmann, Michael Gharbi, Miika Aittala, and Frédo Durand. A dataset of multi-illumination images in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4080–4089, 2019.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

Rohit Pandey, Sergio Orts-Escolano, Chloe Legendre, Christian Haene, Sofien Bouaziz, Christoph Rhemann, Paul Debevec, and Sean Fanello. Total relighting: Learning to relight portraits for background replacement. *ACM Transactions on Graphics*, 40(4), 2021.

Yizuo Peng, Xuelin Chen, Kai Zhang, and Xiaodong Cun. LightCtrl: Training-free controllable video relighting. In *The Fourteenth International Conference on Learning Representations*, 2026. URL `https://openreview.net/forum?id=5ft8vd9rwc`.

Ziran Qin, Youru Lv, Mingbao Lin, Hang Guo, Zeren Zhang, Danping Zou, and Weiyao Lin. Head-aware KV cache compression for efficient visual autoregressive modeling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 24982–24990, 2026. doi: 10.1609/aaai.v40i30.39686.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning*, 2021.

Xinyu Wang, Chongbo Zhao, Fangneng Zhan, and Yue Ma. LiveEdit: Towards real-time diffusion-based streaming video editing. In *European Conference on Computer Vision*, pp. 631–649. Springer, 2026.

Mingqiang Wu, Weilun Feng, Zhefeng Zhang, Haotong Qin, Yuqi Li, Guoxin Fan, Xiaokun Liu, Zhulin An, Libo Huang, Yongjun Xu, and Chuanguang Yang. Echo-forcing: A scene memory framework for interactive long video generation. *arXiv preprint arXiv:2605.16003*, 2026.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=NG7sS51zVF`.

Guangxuan Xiao, Jiaming Tang, Jingwei Zuo, Junxian Guo, Shang Yang, Haotian Tang, Yao Fu, and Song Han. DuoAttention: Efficient long-context LLM inference with retrieval and streaming heads. In *The Thirteenth International Conference on Learning Representations*, 2025.

Weiqing Xiao, Hong Li, Xiuyu Yang, Houyuan Chen, Wenyi Li, Tianqi Liu, Shaocong Xu, Chongjie Ye, Hao Zhao, and Beibei Wang. Relit-live: Relight video by jointly learning environment video. In *ACM SIGGRAPH 2026 Conference Papers*, 2026.

Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, Song Han, and Yukang Chen. Longlive: Real-time interactive long video generation. In *Proceedings of the International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=nCAODkpsPJ.

Yang Yang, Tianyi Zhang, Wei Huang, Jinwei Chen, Boxi Wu, Xiaofei He, Deng Cai, Bo Li, and Peng-Tao Jiang. Anchor forcing: Anchor memory and tri-region RoPE for interactive streaming video diffusion. *arXiv preprint arXiv:2603.13405*, 2026b.

Tianwei Yin, Qiang Zhang, Richard Zhang, William T. Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22963–22974, 2025.

Longwen Zhang, Qixuan Zhang, Minye Wu, Jingyi Yu, and Lan Xu. Neural video portrait relighting in real-time via consistency modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 802–812, 2021. URL https://openaccess.thecvf.com/content/ICCV2021/html/Zhang_Neural_Video_Portrait_Relighting_in_Real-Time_via_Consistency_Modeling_ICCV_2021_paper.html.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Scaling in-the-wild training for diffusion-based illumination harmonization and editing by imposing consistent light transport. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=u1cQYxRI1H.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018.

Zhenyu Zhang, Ying Sheng, Tianyi Zhou, Tianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuandong Tian, Christopher Ré, Clark Barrett, Zhangyang Wang, and Beidi Chen. H2o: Heavy-hitter oracle for efficient generative inference of large language models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34661–34710, 2023. doi: 10.52202/075280-1506.

Yuyang Zhao, Yicheng Pan, Qiyuan He, Jincheng Yu, Junsong Chen, Tian Ye, Haozhe Liu, Enze Xie, and Song Han. SANA-Streaming: Real-time streaming video editing with hybrid diffusion transformer, 2026. URL https://arxiv.org/abs/2605.30409.

Yujie Zhou, Jiazi Bu, Pengyang Ling, Pan Zhang, Tong Wu, Qidong Huang, Jinsong Li, Xiaoyi Dong, Yuhang Zang, Yuhang Cao, Anyi Rao, Jiaqi Wang, and Li Niu. Light-a-video: Training-free video relighting via progressive light fusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13315–13325, 2025.

LLM Usage Disclosure Large language models were used to assist in language polishing, LaTeX formatting, and drafting experimental descriptions. All technical claims, experimental results, and paper content were verified and reviewed manually by the authors, who take full responsibility for the submission. No LLM was used to generate core ideas or experimental designs.

Ethics Statement All human user studies were conducted with informed consent from participants. Participants were compensated fairly. We only process publicly available video datasets; no private or sensitive personal data was collected in our experiments. The proposed video relighting method does not create deepfakes and is intended for post-production and virtual production research purposes.

A Appendix

This appendix follows the order of the experimental results in the main paper. We first describe how the paired relighting data are constructed, then present additional quantitative and qualitative comparisons, report the complete perceptual evaluation, and finally analyze head allocation, attention-source design, the global head budget, and the routing–recovery interaction.

A.1 Data Construction

Source videos and split. Figure 6 summarizes the pipeline. We collect 2,500 source candidates and retain 1,000 videos after filtering and deduplication. These videos are split by identity into 800 training videos and 200 test videos, so that no identity appears on both sides. The short clips, medium clips, and interactive streams in Sec. 4.1 are drawn from the 200 held-out videos. MIT Multi-Illumination is a separate public benchmark and is not part of this split.

Paired targets. For the 800 training videos only, we synthesize targets conditioned on an HDR environment map, a reference image, or a text lighting instruction, using Wan2.1-VACE (Jiang et al., 2025) and Editto (Bai et al., 2026). A candidate is kept only when the target preserves the source content, remains temporally consistent, and follows the requested light; candidates that fail any check are discarded. Figure 7 shows retained pairs: within each row the scene and motion follow the source, while the illumination changes with the conditioning modality.

Model-assisted refinement. The accepted pairs train an initial constant-light relighting model. We then run two refinement rounds. In each round the model regenerates the targets that failed the checks, under newly sampled conditions, and the same three filters are applied again. Pairs that still fail are dropped. The final training set contains the 22,000 pairs that pass all three checks: 6,300 HDR, 6,100 reference-image, and 9,600 text-conditioned. The text pairs include free-form lighting instructions and continuous parameter control, where color and intensity are written into the prompt and encoded by E_{param}.

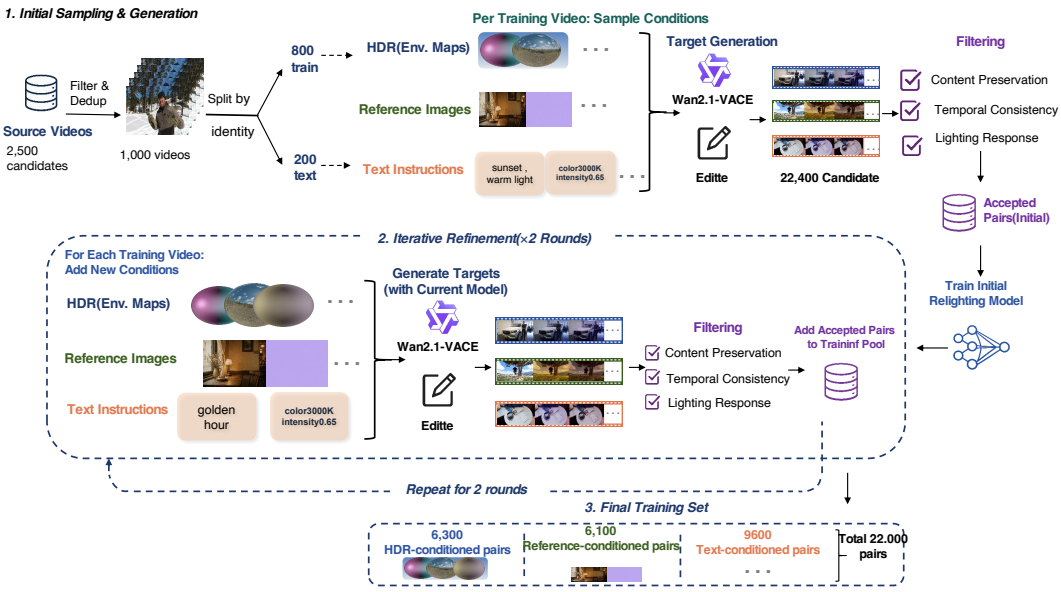

Figure 6: Data construction and training pipeline. Source candidates are filtered to 1,000 videos and split by identity. Synthesized targets are filtered for content, motion, and lighting. Failed targets are regenerated in two rounds. The final training set contains 22,000 pairs from the 800 training videos. Text inputs include continuous color and intensity.

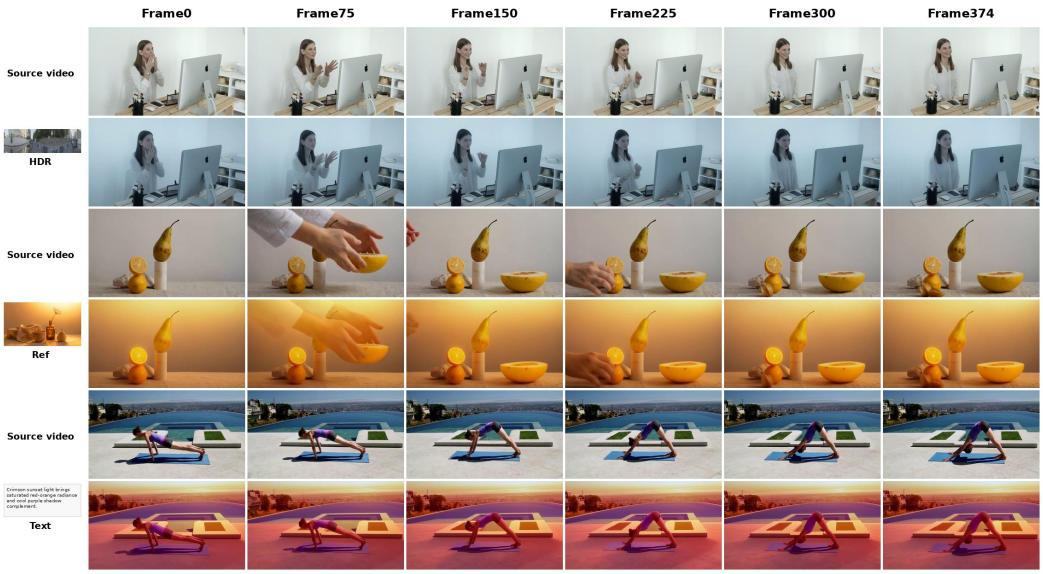

Figure 7: Representative pairs from the constructed dataset. Rows show a source video and the HDR-, reference-, and text-conditioned targets. Columns sample frames from 0 to 374.

A.2 Evaluation Metrics

Notation and frame statistics. Let $\hat{y}_t$ be the output frame at time t with RGB values in $[0, 1]$, x_t the source frame, f the frame rate, and t_{sw} the first frame after the edit. All statistics use the central crop Ω covering 55% of the frame height and 55% of the width, and luminance $Y = 0.299R + 0.587G + 0.114B$. For a frame y we compute

$$\beta(y) = \mathrm{mean}_\Omega\, Y, \quad \kappa(y) = \log \frac{\mathrm{mean}_\Omega\, R}{\mathrm{mean}_\Omega\, B}, \quad s_x(y) = \frac{Y_\mathrm{r} - Y_\mathrm{l}}{Y_\mathrm{r} + Y_\mathrm{l}}, \quad s_y(y) = \frac{Y_\mathrm{b} - Y_\mathrm{t}}{Y_\mathrm{b} + Y_\mathrm{t}}, \tag{11}$$

i.e., brightness, a red–blue color-temperature proxy, and horizontal and vertical shading asymmetry, where $Y_\mathrm{l}, Y_\mathrm{r}, Y_\mathrm{t}, Y_\mathrm{b}$ are the mean luminance of the left, right, top, and bottom halves of Ω. The image-plane light direction is $\varphi(y) = \mathrm{atan2}(s_y, s_x)$, and $\Delta\varphi$ denotes the wrapped angular difference in degrees.

Light estimator. For each pixel p in the central crop, public models predict a unit normal n_p and an RGB shading $S(p) \in \mathbb{R}^3$. Let $Y(n) \in \mathbb{R}^{16}$ be the real spherical harmonics of order $l \le 3$. The light of a frame y is the least-squares fit

$$\hat{L}(y) = \arg\min_{L \in \mathbb{R}^{16\times 3}} \sum_{p\in\Omega} \left\| S(p) - Y(n_p)^\top L \right\|_2^2. \tag{12}$$

Requested light per modality. Each condition provides an SH light L^* and targets $(\beta^*, \kappa^*, \varphi^*)$:

- *HDR*: L^* is the third-order SH projection of the environment map E; β^* is its mean luminance, κ^* the log red–blue ratio of its luminance-weighted mean color, and φ^* the direction given by the $l{=}1$ SH coefficients.
- *Reference image*: $L^* = \hat{L}(I_{\mathrm{ref}})$, and $\beta^*, \kappa^*, \varphi^*$ are computed from I_{ref} with Eq. (11).
- *Text*: a rule-based keyword parser (no language model) reads the prompt. *left/right/top/bottom* (and synonyms such as *overhead*, *below*) set φ^*; *bright/strong/dramatic* set $\beta^*{=}0.85$, *dim/soft/weak/subtle* set $\beta^*{=}0.2$, and $\beta^*{=}0.5$ otherwise; a color word (red, orange, yellow, green, cyan, blue, purple, warm, cool) maps to a fixed RGB triplet that gives κ^*. L^* is the third-order SH projection of a directional light of that color along $\omega^* = (\cos\varphi^*, \sin\varphi^*, 1)/\sqrt{2}$, in the image axes of Eq. (11) with z toward the camera. A missing direction defaults to $\omega^* = (0, 0, 1)$ and a missing color to white.

- *Parameter control*: θ_t is written into a text instruction. LF still uses the keyword protocol above, so a continuous RGB or intensity value is scored only when the parser maps it to a color word and one of the three brightness levels.

LF. LF is evaluated on frames sampled every 2 s. Bars denote the mean of a statistic over the sampled frames. For an SH light L, let $R_L(n, c)$ be the irradiance of a unit-albedo Lambertian sphere at normal n and color channel c, evaluated on the set $\mathcal{N}$ of camera-facing normals, and let $\tilde{R}_L = R_L / \operatorname{mean}_{n,c} R_L$ remove its global brightness scale. Then

$$\mathrm{LF} = 100 \cdot \overline{\operatorname{mean}_{n \in \mathcal{N},\, c} \big| \tilde{R}_{\hat{L}(\hat{y}_t)}(n, c) - \tilde{R}_{L^*}(n, c) \big|}. \tag{13}$$

Lower is better. Because each rendering is divided by its mean, a global brightness scale is removed: LF compares normalized lighting appearance within a modality and does not measure absolute intensity. The fitted $\hat{L}$ lives in the shading basis. HDR L^* is an incident-radiance projection and is not the same coefficient type; the two are comparable only after the incident coefficients are converted to irradiance by the Lambertian cosine convolution. Because L^* is derived differently for each modality, LF is compared only within a modality.

Switch targets. For light $u \in \{\mathrm{A}, \mathrm{B}\}$, let $\hat{y}_t^u$ be the output of the evaluated method on the same source video and noise under the constant light c^u, and $b_u(t) = 255\, \beta(\hat{y}_t^u)$; Jump and T_{90} use these constant-light references. OldRes compares the output with the requested lights: the anchor $\mathbf{m}^u = (\beta^u, \kappa^u, s_x^u, s_y^u)$ is derived from the condition as above: from the environment map for HDR, from Eq. (11) on the reference image for reference control, and from the parsed prompt for text, with $(s_x, s_y) = (\cos\varphi^*, \sin\varphi^*)$. With $\mathbf{m}(\hat{y}_t)$ the same four statistics of the output, let $d_u(t) = \|\mathbf{m}(\hat{y}_t) - \mathbf{m}^u\|_2$.

OldRes. Over the first second after the edit, $\mathcal{P} = \{t_{\mathrm{sw}}, \ldots, t_{\mathrm{sw}} + f - 1\}$,

$$\mathrm{OldRes} = \frac{1}{|\mathcal{P}|} \sum_{t \in \mathcal{P}} \frac{d_{\mathrm{B}}(t)}{d_{\mathrm{A}}(t) + d_{\mathrm{B}}(t)}, \tag{14}$$

which is 0 when the output matches the new light and 1 when it matches the old light.

Jump. With $b_{\mathrm{out}}(t) = 255\, \beta(\hat{y}_t)$, the output change $\Delta_{\mathrm{out}} = b_{\mathrm{out}}(t_{\mathrm{sw}}) - b_{\mathrm{out}}(t_{\mathrm{sw}} - 1)$, and the correct switch $\Delta^* = b_{\mathrm{B}}(t_{\mathrm{sw}}) - b_{\mathrm{A}}(t_{\mathrm{sw}} - 1)$,

$$\mathrm{Jump} = |\Delta_{\mathrm{out}} - \Delta^*|, \tag{15}$$

the luminance change at the edit beyond the correct switch, in 8-bit units.

T_{90}. Let L_0 be the mean of b_{out} over the 0.5 s before the edit and L_{ss} the mean of b_{B} over the settled window $[t_{\mathrm{sw}} + 5\,\mathrm{s},\ t_{\mathrm{sw}} + 8\,\mathrm{s}]$. With the progress $\pi(t) = (b_{\mathrm{out}}(t) - L_0)/(L_{\mathrm{ss}} - L_0)$,

$$T_{90} = \min\Big\{ \tfrac{t - t_{\mathrm{sw}}}{f} :\ t \geq t_{\mathrm{sw}},\ \pi(t') \geq 0.9 \text{ for } t \leq t' < t + 9 \Big\}, \tag{16}$$

i.e., the time after the edit at which the output reaches 90% of the new-light luminance change and holds it for nine frames. If this never holds, T_{90} is the remaining clip length. Streams with $|L_{\mathrm{ss}} - L_0| < 10^{-3}$ are excluded.

SwID. With the DINOv2 ViT-B/14 (Oquab et al., 2024) class-token embedding $e(\cdot)$ of the central crop,

$$\mathrm{SwID} = \frac{1}{12} \sum_{i=1}^{3} \sum_{j=0}^{3} \cos\big(e(\hat{y}_{t_{\mathrm{sw}} - i}),\, e(\hat{y}_{t_{\mathrm{sw}} + j})\big), \tag{17}$$

the mean similarity between the three frames before and the four frames from the edit onward.

Source preservation. DINO, $\mathrm{LPIPS}_{\mathrm{src}}$, and $\mathrm{CLIP}_{\mathrm{inter}}$ are evaluated on frames sampled every 2 s, and bars denote the mean over these frames. With $e(\cdot)$ the DINOv2 embedding used for SwID and LPIPS (Zhang et al., 2018) computed on Ω,

$$\mathrm{DINO} = \overline{\cos\big(e(\hat{y}_t),\, e(x_t)\big)}, \qquad \mathrm{LPIPS}_{\mathrm{src}} = \overline{\mathrm{LPIPS}\big(\hat{y}_t,\, x_t\big)}, \tag{18}$$

so DINO rewards and $\mathrm{LPIPS}_{\mathrm{src}}$ penalizes departures from the source content and structure. With $e_{\mathrm{C}}(\cdot)$ the CLIP (Radford et al., 2021) image embedding of Ω,

$$\mathrm{CLIP}_{\mathrm{inter}} = \overline{\cos\big(e_{\mathrm{C}}(\hat{y}_t),\, e_{\mathrm{C}}(\hat{y}_{t+2f})\big)}, \tag{19}$$

the appearance similarity between frames 2 s apart.

Temporal metrics. tOF and Flicker use anchor frames t sampled every 2 s, but each anchor is compared with its immediate predecessors $t-1$ and $t-2$ at the native frame rate, so both metrics measure frame-to-frame changes; sparse anchors only reduce cost. Let $\mathcal{W}_{t,k}$ warp frame $t-k$ to frame t using Farneback flow estimated on the *source* frames, with frames resized to a long side of 512 pixels, and let $\mathcal{V}_t$ be the pixels in Ω that pass the forward–backward occlusion check for both $k=1$ and $k=2$. Then

$$\mathrm{tOF} = \overline{\mathrm{mean}_{\Omega}\big|\hat{y}_t - \mathcal{W}_{t,2}(\hat{y}_{t-2})\big|}, \tag{20}$$

$$\mathrm{Flicker} = 100 \cdot \overline{\mathrm{mean}_{\mathcal{V}_t}\big|Y(\hat{y}_t) - 2\,\mathcal{W}_{t,1}\big(Y(\hat{y}_{t-1})\big) + \mathcal{W}_{t,2}\big(Y(\hat{y}_{t-2})\big)\big|}, \tag{21}$$

where tOF is the RGB residual on the central crop and Flicker is the second temporal difference of aligned luminance on its non-occluded pixels, which removes smooth brightness trends and keeps high-frequency changes. Because the flow comes from the source, both metrics penalize output changes that source motion does not explain, such as flickering shading or drifting color, and a method cannot lower them by following its own motion instead of the source.

A.3 Additional Quantitative Comparison

Paired-image relighting. Table 4 supplements the fixed-light comparison in Sec. 4.2 with paired-target evaluation on MIT Multi-Illumination. PSNR, $\mathrm{SSIM}_{\mathrm{GT}}$, and $\mathrm{LPIPS}_{\mathrm{GT}}$ measure agreement with the ground-truth image, while LF measures agreement with the target illumination. Together, these metrics evaluate image reconstruction and lighting accuracy.

CoRe-HDR obtains the highest PSNR and $\mathrm{SSIM}_{\mathrm{GT}}$ and the lowest $\mathrm{LPIPS}_{\mathrm{GT}}$ and LF among the compared methods, with Relit-LiVE second on every column. These results extend the video comparison to a benchmark with paired ground-truth illumination.

Method	PSNR↑	$\mathrm{SSIM}_{\mathrm{GT}}$↑	$\mathrm{LPIPS}_{\mathrm{GT}}$↓	LF↓
RelightVid (2025)	16.24	0.598	0.342	7.82
Relit-LiVE (2026)	<u>21.86</u>	<u>0.849</u>	<u>0.132</u>	<u>5.88</u>
Neural Gaffer (2024)	17.87	0.683	0.241	6.74
Diff. Renderer (2025b)	17.29	0.622	0.355	7.05
UniRelight (2025)	20.76	0.749	0.251	6.18
UniLumos (2025a)	18.42	0.698	0.276	7.15
CoRe-HDR	**22.52**	**0.871**	**0.130**	**5.69**

Table 4: Paired-GT evaluation on MIT Multi-Illumination stills. Best and second-best results are bold and underlined, respectively. LF uses the same definition as Table 1. Across the seven method-level means shown, the Spearman correlation between PSNR and LF is $\rho = -0.89$.

A.4 Additional Qualitative Results

The following examples extend the qualitative comparison in Sec. 4.2. We present text and reference conditioning, paired-image relighting, and streaming sequences with repeated HDR edits.

Text-Conditioned Relighting

Figure 8 compares methods at corresponding frames under the same text condition. The example examines both the requested lighting appearance and the preservation of the source subject and scene. CoRe applies warm illumination while retaining the source scene and the subject's appearance across the displayed frames. The sequence also shows how the lighting effect persists as the subject moves.

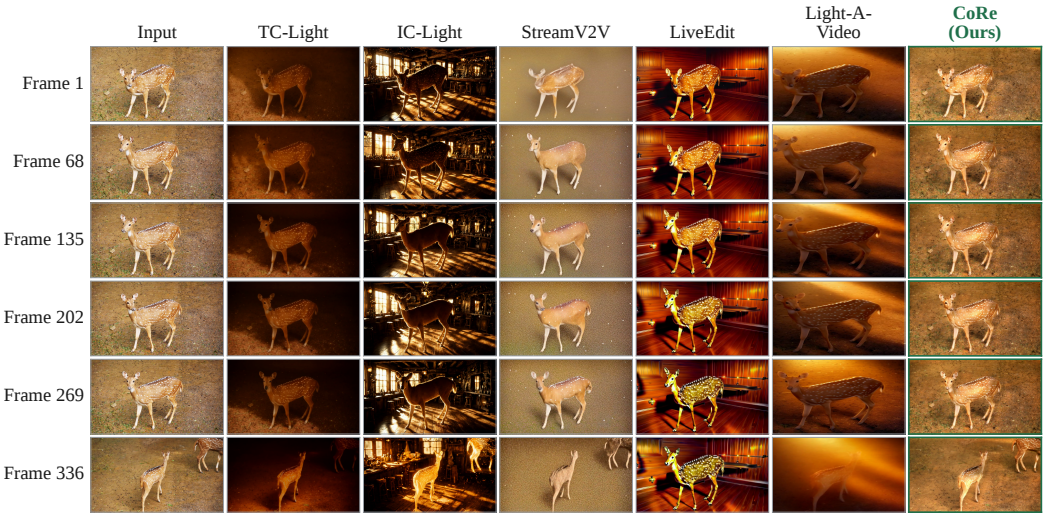

Figure 8: Text-conditioned qualitative comparison. CoRe matches the requested lighting description without collapsing identity or motion.

Reference-Conditioned Relighting

Figure 9 evaluates lighting transfer from a reference condition. The requested lighting appearance is shown in the source-frame insets. CoRe produces the pink and green illumination across the person, vehicle, and surrounding scene while preserving their spatial arrangement. The displayed frames show the lighting response at multiple points in the sequence, complementing the text-conditioned comparison.



Figure 9: Reference-conditioned qualitative comparison. CoRe transfers the lighting appearance of the reference image while keeping scene content and identity.

A.5 Continuous Parameter Control

Parameter conditioning. Continuous control specifies the current light color and intensity through a numerical parameter vector θ_t, written into a text lighting instruction. The text encoder and projection, denoted E_{param} and P_{param} in this mode, map that instruction to the shared light-token interface,

$$Z_t = P_{\text{param}}\big(E_{\text{param}}(\theta_t)\big). \tag{22}$$

At each video-block boundary, the current settings update the light tokens and light K/V. CoRe retains the video history and uses its fixed head assignment throughout the parameter trajectory.

Qualitative comparison. Figure 10 shows a sequence in which light color changes while intensity increases. CoRe produces orange, yellow, cyan, and blue lamp illumination at the selected frames,



Figure 10: Qualitative results of continuously controllable video relighting on the boat sequence. Light color and intensity vary continuously over time. Rows show the input, LiveLight, and CoRe, respectively. Columns correspond to frames 78, 121, 192, and 218.



Figure 11: MIT Multi-Illumination qualitative comparison on two scenes. Columns: target environment map, input, HDR-conditioned baselines, Ours, and GT. All methods are conditioned on the same environment map. Ours follows the GT brightness and color while preserving scene structure.

with corresponding changes on nearby surfaces. LiveLight produces broader color changes across the sky, buildings, and water, whereas CoRe's changes are more concentrated around the lamp.

Method	LF↓	DINO↑	$\text{LPIPS}_{\text{src}}$ ↓	tOF↓	Flk.↓	$\text{CLIP}_{\text{inter}}$ ↑
LiveLight	**6.582**	0.900	0.261	0.0126	2.252	0.928
CoRe	6.864	**0.936**	**0.120**	**0.0118**	**2.086**	**0.977**

Table 5: Quantitative comparison under continuous parameter control. $\text{CLIP}_{\text{inter}}$ measures cross-frame appearance consistency. Best results are bold.

Quantitative comparison. Table 5 compares lighting error, source preservation, and temporal stability. CoRe has stronger source preservation and lower tOF and Flicker. LiveLight has lower LF. Under a continuously changing light, $\text{CLIP}_{\text{inter}}$ is not read as a fidelity score.

Paired-Target Qualitative Comparison

Figure 11 complements the numerical results in Table 4. The source and ground-truth images establish the requested appearance change, and the method outputs show how each approach reproduces the target illumination. The comparison allows direct inspection of brightness, color, shading, and scene structure alongside the paired-image metrics.

Repeated HDR Lighting Edits

Figure 12 presents streaming results with successive HDR lighting requests. Each source row includes the environment map active at the displayed frame, and the corresponding CoRe row shows

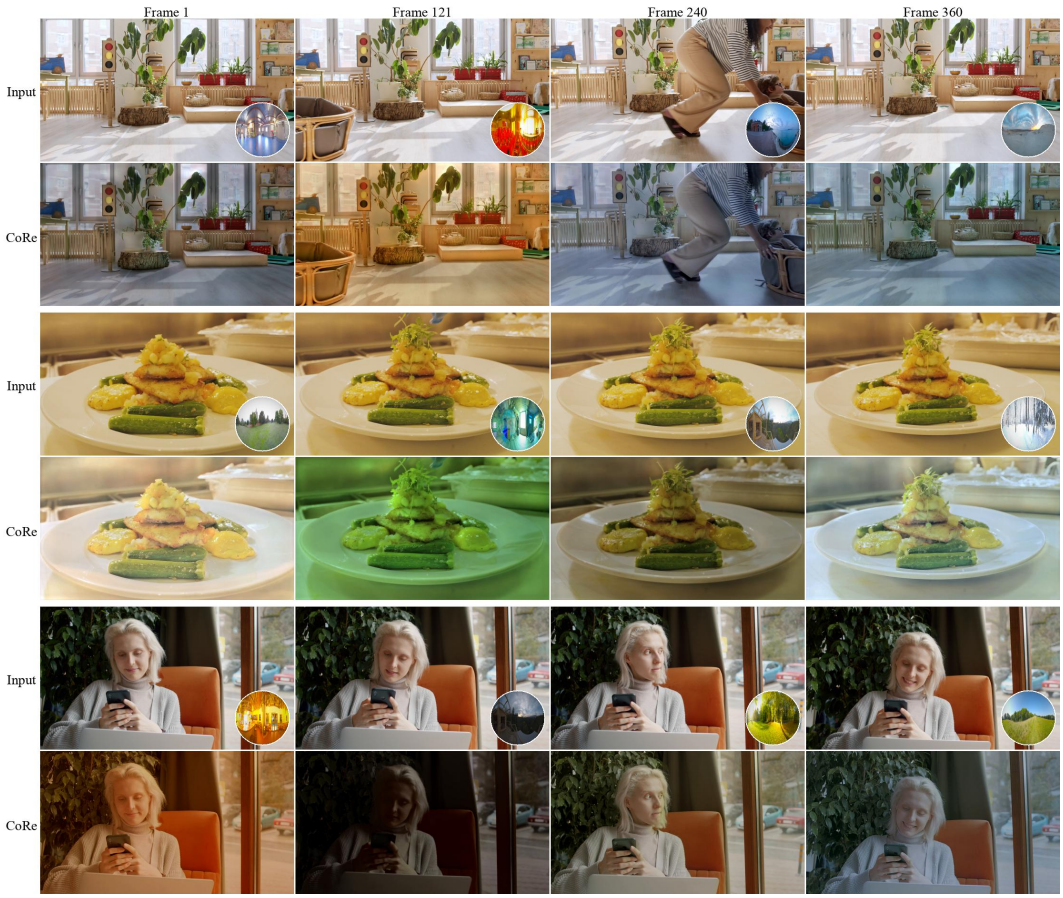

Figure 12: Streaming relighting with repeated HDR edits. For each scene, the input row shows source frames with the requested environment map (inset), which changes during the stream; the CoRe row shows the relit output at frames 1, 121, 240, and 360.

the relit output. Frames 1, 121, 240, and 360 sample the evolving lighting conditions and scene content.

CoRe uses the same head assignment throughout each stream. Updated light tokens supply the current-light heads, while video-memory heads retain the rolling context across edits. The examples show repeated changes in lighting color and intensity as the source scene continues to evolve, extending the single-edit quantitative evaluation in the main paper.

A.6 Human Perceptual Evaluation

Following Sec. 4.2, 50 participants compare CoRe with a baseline for the same source content and target lighting condition under anonymized and randomized presentation. The evaluation covers visual realism (VR), physical consistency (PC), and lighting alignment (LA). VR concerns the overall plausibility of the result, PC concerns the consistency of illumination with scene content, and LA concerns agreement with the requested light. Every reported value is the percentage of pairwise trials preferring CoRe over the named method.

Short, Medium, and Long Videos

Table 6 consolidates the video comparisons by conditioning modality and duration. CoRe receives majority preference in every evaluated video setting. The short and medium comparisons cover both text- and HDR-conditioned baselines, while the long-stream comparison evaluates text-conditioned

Model	Condition	Short (5–10 s)			Medium (10–30 s)			Long switch (45.6–60 s)		
		VR↑	PC↑	LA↑	VR↑	PC↑	LA↑	VR↑	PC↑	LA↑
TC-Light (2025b)	Text	58.6	61.8	59.7	56.7	60.8	54.9	–	–	–
Light-A-Video (2025)	Text	62.4	65.2	63.8	61.8	64.5	62.1	–	–	–
IC-Light (2025)	Text	89.7	91.2	87.9	89.5	91.3	88.0	–	–	–
LiveEdit (2026)	Text	68.9	72.5	70.8	66.9	71.4	69.2	62.4	77.3	81.5
StreamV2V (2025a)	Text	55.8	60.9	62.4	71.8	74.6	72.1	71.6	82.8	85.2
RelightVid (2025)	HDR	77.9	82.1	85.4	–	–	–	–	–	–
Relit-LiVE (2026)	HDR	68.4	70.2	72.6	–	–	–	–	–	–
Neural Gaffer (2024)	HDR	61.4	72.8	74.6	64.8	73.5	75.1	–	–	–
Diffusion Renderer (2025b)	HDR	70.6	76.2	78.5	–	–	–	–	–	–
UniRelight (2025)	HDR	65.8	70.9	72.7	–	–	–	–	–	–
UniLumos (2025a)	HDR	72.5	76.8	78.2	70.2	74.6	76.8	–	–	–
LongLive + Adapter (2026a) (F_0)	Text	–	–	–	–	–	–	57.2	62.4	64.8
LongLive + Recache	Text	–	–	–	–	–	–	54.6	58.8	61.2

Table 6: Consolidated user study across short, medium, and long videos. Each model-condition pair occupies one row; dashes denote settings not evaluated. Each entry is the percentage of pairwise trials preferring Ours. VR, PC, and LA denote visual realism, physical consistency, and lighting alignment.

editing. On long streams the margin is largest on lighting alignment. These judgments complement the lighting-response and temporal metrics reported for interactive streams.

MIT Multi-Illumination

Table 7 reports pairwise preferences on still images. CoRe receives majority preference for physical consistency and lighting alignment against every listed baseline. Against Relit-LiVE and UniLumos, visual realism is slightly preferred the other way. The paired-image study therefore captures differences in visual preference alongside the reconstruction and lighting metrics in Table 4.

Model	Condition	VR↑	PC↑	LA↑
RelightVid (2025)	HDR	62.8	66.4	68.5
Relit-LiVE (2026)	HDR	48.2	51.4	52.6
Neural Gaffer (2024)	HDR	62.4	66.8	69.2
Diff. Renderer (2025b)	HDR	61.5	65.7	67.2
UniRelight (2025)	HDR	52.6	56.8	59.5
UniLumos (2025a)	HDR	48.7	54.2	56.8

Table 7: User study on MIT Multi-Illumination stills. Each entry is the percentage of pairwise trials preferring CoRe. VR, PC, and LA denote visual realism, physical consistency, and lighting alignment.

Commercial Video Generators

Table 8 reports the reference-image comparison with commercial video generators using the same three criteria. CoRe receives majority preference against all four models. The margin is largest against Kling V3 and smallest against MiniMax. These results complement the commercial-model qualitative comparison in Fig. 4.

Model	VR↑	PC↑	LA↑
Wan 3.0	74.8	76.2	79.5
MiniMax	61.3	63.8	68.4
Kling V3	87.6	85.1	89.2
Seedance 2.5	69.4	72.7	71.8

Table 8: Commercial generators under reference-image control.

A.7 Additional Ablation Analysis

This section follows the ablation analysis in Sec. 4.3. We first examine how head identities and layer counts affect recovery, then compare attention-source designs, and finally analyze the global head budget.

CAUSAL PATCHING

Figure 13 separates the two head groups used in the introduction. Sensitive heads carry more old light when their K/V are replaced by the previous illumination. Insensitive heads lose more geometry when their K/V are replaced by another clip.

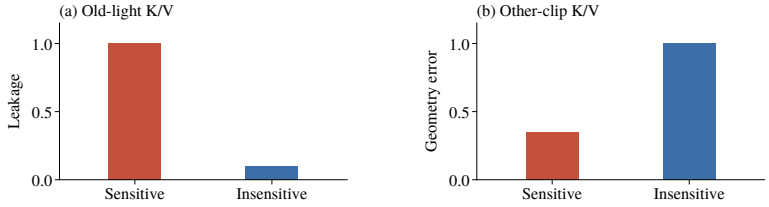

Figure 13: Causal patching of sensitive and insensitive heads. (a) Injecting old-light K/V. (b) Swapping in another clip's K/V. Panel (b) shows only the direction of the effect.

HEAD IDENTITIES AND LAYER COUNTS

Table 9 compares allocation rules on HDR streams. The first three configurations assign three light-control heads to each routed layer and vary their identities. Uniform+Random samples head identities, Uniform+Sensitivity selects them using the lighting-swap probe, and Uniform+Learned optimizes their identities through training. CoRe jointly learns the head identities and their distribution across layers under the same global budget $B = 87$.

With fixed layer counts, sensitivity-based selection already lowers OldRes and T_{90} relative to a random partition. Learning the identities improves both further, and jointly learning the layer counts gives the lowest OldRes and T_{90} together with the highest SwID. The progression shows the contribution of learning both which heads read the current light and how those heads are distributed across layers.

Variant	Counts	Identities	OldRes↓	T_{90} (s)↓	SwID↑	FPS↑
Uniform+Random	Frozen	Random	0.24±0.03	2.88±0.22	0.964±0.004	25.4
Uniform+Sensitivity	Frozen	Probe	0.15	2.18	0.968	25.4
Uniform+Learned	Frozen	Learned	<u>0.11</u>	<u>1.72</u>	0.971	25.4
Concat-all	All	All	0.17	2.05	0.970	20.8
Augment-mask	CoRe	CoRe	0.16	1.96	<u>0.972</u>	20.9
CoRe	Learned	Learned	**0.05**	**0.72**	**0.973**	25.4

Table 9: Ablations on HDR streams. All variants use $B = 87$ except Concat-all (348 heads). The first three fix three light heads per layer; Augment-mask concatenates video/light K/V on CoRe-selected heads. FPS measures generator-only throughput at 832×480. For Uniform+Random, OldRes, T_{90}, and SwID are reported as mean±standard deviation across eight random head partitions. Bold/underline indicate best/second-best values.

ATTENTION-SOURCE REPLACEMENT AND AUGMENTATION

The Concat-all and Augment-mask rows in Table 9 compare alternative uses of the current light tokens. Concat-all appends light K/V to the video K/V of all heads. Augment-mask applies this operation to the head set selected by CoRe. In both variants, the affected heads retain access to the cached video history. CoRe assigns its selected heads to current-light K/V, with video-derived queries.

Concat-all and Augment-mask leave a larger OldRes and a longer T_{90} than CoRe, and their generator throughput stays close to that of F_0. Replacement on the selected heads lowers both residuals and raises throughput, because those heads no longer attend to the video window. These results connect the choice of attention source to both lighting recovery and inference cost.

GLOBAL HEAD BUDGET

Table 10 evaluates budgets $B = 58$, 87, and 116. For each budget, the frozen-count control assigns $k_\ell = B/29$ heads to the light route in every layer. CoRe learns the allocation across layers while preserving the same total number of light-control heads.

Learning the allocation lowers OldRes at every tested budget. Across the three CoRe settings, a larger budget reduces cache memory and increases throughput, with a small drop in SwID. OldRes is lowest at $B = 87$ (0.05), below both $B = 58$ (0.13) and $B = 116$ (0.07), so $B = 87$ is the default: it gives the best lighting recovery while keeping identity, throughput, and memory close to the other budgets.

B	Allocation	k_ℓ	OldRes↓	SwID↑	FPS↑	Mem↓
–	F_0	0	0.19	0.976	21.0	20.8
58	Frozen k_ℓ	2	0.17	**0.975**	23.7	20.2
58	**CoRe**	Learned	**0.13**	0.974	23.7	20.2
87	Frozen k_ℓ	3	0.11	0.971	25.4	19.6
87	**CoRe**	Learned	**0.05**	**0.973**	25.4	19.6
116	Frozen k_ℓ	4	0.10	0.969	27.3	18.9
116	**CoRe**	Learned	**0.07**	**0.971**	27.3	18.9

Table 10: Global head-budget comparison. Frozen counts use $k_\ell = B/29$ in each layer; F_0 routes no heads. Bold indicates the better value within each budget. FPS measures generator-only throughput at 832×480; Mem is peak GPU memory in GB.

ROUTING AND RECOVERY SUPERVISION

Table 11 crosses head routing with recovery supervision on the same HDR streams. The no-routing, no-recovery cell is F_0. The routing-plus-recovery cell is CoRe at B=87. Recovery supervision is $\mathcal{L}_{\text{sw}} + \mathcal{L}_{\text{rec}}$; every cell still uses $\mathcal{L}_{\text{rel}}$.

Either change lowers OldRes and T_{90}. The drop is larger when the other factor is already present: recovery lowers OldRes by 0.03 without routing and by 0.09 with it, and routing lowers it by 0.05 without recovery and by 0.11 with it. T_{90} shows the same pattern.

Routing	Recovery	OldRes↓	T_{90} (s)↓
No	No	0.19	2.52
No	Yes	0.16	2.14
Yes (B=87)	No	0.14	1.92
Yes (B=87)	Yes	**0.05**	**0.72**

Table 11: Routing × recovery on HDR streams.