LOCO: Local Light-Aware Object Compositing with Spatially Varying Illumination-Augmented Data

Jinseo Jeong[1], Hyunsoo Kim[1], Junseo Koo[1], Junhyeog Yun[1], Gunhee Kim[1], Soo Ye Kim[2]*
[1]Seoul National University [2]Adobe Research

Project page: `https://jinseo.kr/LOCO`

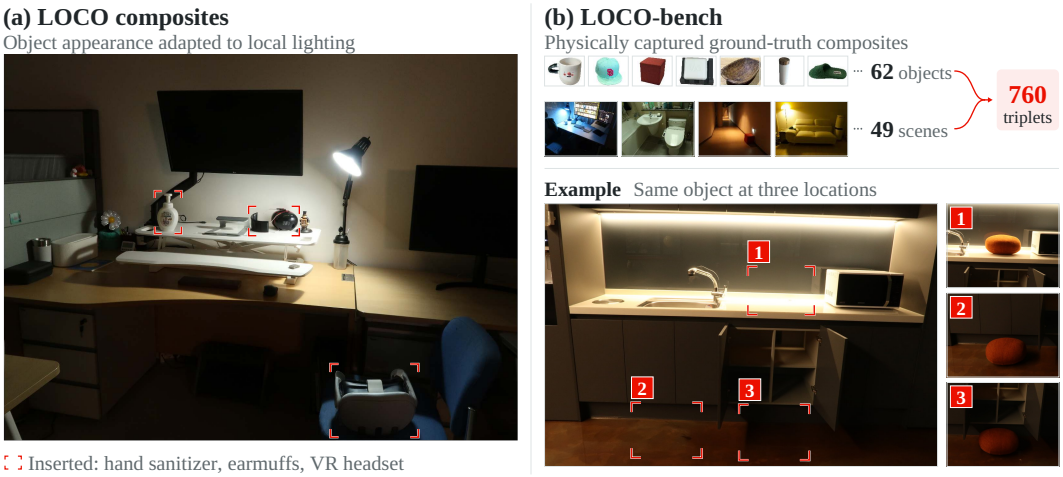


Figure 1: **Object compositing under spatially varying illumination.** *(a)* Object compositing inserts a user-provided object into a background image at a chosen location. Within a single scene, surface brightness, color cast, and shadow direction vary with location—brighter near light sources and dimmer farther away. Our method, LOCO, adapts the inserted object's appearance to the local lighting at the inserted location. *(b)* LOCO-bench provides physically captured ground-truth composites of the same object at different locations within a shared scene. Numbered boxes mark the insertion locations; crops show the captured ground truth at each.

Abstract

In indoor scenes, light sources and occlusions from scene geometry create *spatially varying illumination*, including brightness gradients, cast shadows, and local color shifts that shape how every object appears. A less explored aspect of diffusion-based object compositing is whether an inserted object can adapt to this local illumination at the desired location. The core challenge is the scarcity of large-scale training data capturing how object appearance should adapt to location-dependent illumination. We present LOCO, a data augmentation framework for LOcal light-aware object COmpositing that leverages publicly available monocular videos as illumination sources. We treat each video frame as a virtual spotlight anchored at its camera pose, so that viewpoint variation across a video yields diverse light directions. With controllable beam shape, color, and intensity, this design enables large-scale generation of training data for placement-dependent illumination *on the fly*. We also introduce LOCO-bench, an indoor benchmark of physically captured composites under spatially varying illumination, where the same object is placed at

*Corresponding author.

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

multiple locations within each scene. Experiments on Laval Indoor SV HDR and LOCO-bench show that LOCO outperforms prior methods in local illumination adaptation while preserving object identity.

1 Introduction

Within a single indoor scene, illumination varies from point to point. A tabletop near a window may sit in bright daylight while a far corner falls into shadow [Li et al., 2020, Garon et al., 2019, Srinivasan et al., 2020].

For an inserted object to appear naturally in the scene, its overall brightness, gradient lighting, and cast shadow must match the illumination at the chosen location. Although this is an important aspect of object compositing, prior methods largely overlook this lighting effect.

Diffusion-based methods [Ho et al., 2020, Esser et al., 2024] have advanced object compositing in several directions, including geometric alignment [Song et al., 2023, Chen et al., 2024a, Yu et al., 2025a], semantic coherence [Tewel et al., 2025, Lu et al., 2023, Song et al., 2026, Huang et al., 2025], and color harmonization [Wang et al., 2023, Chen et al., 2024c]. Some recent approaches further improve global appearance alignment by leveraging diffusion-generated training data [Huang et al., 2025, Wang et al., 2025a].

However, lighting variation in their training data remains scene-wide rather than position-specific within a scene—each training image still carries a single global lighting state, providing no signal for how an inserted object's appearance should depend on its particular location. Fig. 2 illustrates this limitation: the inserted object's appearance does not adapt appropriately to different placements, even when local illumination differs substantially across locations within the same scene.

The main obstacle for making compositing models lighting-aware is the lack of training data that link object appearance to location-dependent lighting at scale. Two aspects of current dataset construction pipelines make such data difficult to obtain at scale (Fig. 3).

First, capturing structured, spatially varying illumination at training scale is costly. Real-world collection typically requires carefully staged scenes with repeatedly placing the same object across multiple locations, limiting the scale of data that can be collected. Synthetic alternatives based on game engines [Community, 2018, Epic Games, 2024] offer controllable lighting, but their scale is similarly constrained by the cost of authoring complex scenes. Diffusion-based data generation methods [Huang et al., 2025] reduce manual scene setup, yet offer limited control over illumination and often require substantial human filtering.

Second, commonly used segmentation-derived datasets fail to preserve the local shading context during object insertion. Datasets such as SA-Co [Carion et al., 2025] and OpenImages [Kuznetsova et al., 2020] simulate insertion by masking foreground regions and filling the resulting holes. This process removes local shading cues, including adjacent surfaces and ground-plane shading that reveal the lighting at the insertion location.



Figure 2: **Same object inserted at three locations within one scene.** A DreamBooth [Ruiz et al., 2023] object is placed at three locations in a Laval Indoor SV HDR [Garon et al., 2019] scene, where local illumination varies across insertion location. Across all four existing methods, surface brightness and shadow direction show limited variation across the three locations, indicating that current models respond only weakly to insertion location.

	AnyDoor [Chen et al., 2024a], AnyInsertion [Song et al., 2026]	ORIDa [Kim et al., 2025]	ObjectMover [Yu et al., 2025a]	DreamFuse [Huang et al., 2025]	LOCO
Data acquisition	Segmentation	Photo capture	Game engine	Model generation	Public videos
Data domain	Real	Real	Synthetic	Synthetic	Real
On-the-fly light control	✗	✗	✗	✗	✓
Human-free labeling	✓	✗	✗	✗	✓
Spatial illumination	✗	✗	△	✗	✓

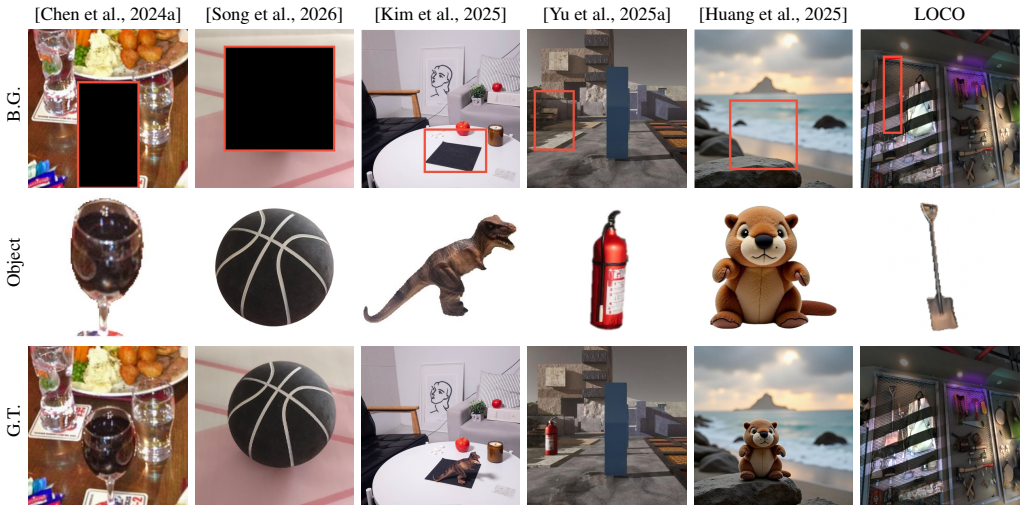


Figure 3: Comparison of data construction methods for object compositing. Prior approaches build training data from segmentation datasets or staged scenes, where spatially varying illumination is rarely captured; ObjectMover supports offline control of a global environment light, so its placement-dependent shading is driven primarily by geometric occlusion. LOCO instead augments real-world videos with online, multi-source spatially varying illumination.

To overcome these limitations, we introduce LOCO, a physically grounded data augmentation framework for **LO**cal light-aware object **CO**mposition that leverages monocular videos as illumination sources. Our idea is to reinterpret each video frame as a pose-anchored virtual spotlight, so that camera motion yields diverse light directions. With controllable beam shape, color, and intensity, this design produces placement-dependent illumination via physically based rendering (PBR) over the recovered scene geometry. Since illumination is generated on-the-fly rather than precomputed, the pipeline scales naturally as data augmentation during training. The pipeline can be applied to any monocular video of a static scene, and we instantiate it on publicly available datasets (uCO3D [Liu et al., 2025], DL3DV-10K [Ling et al., 2024], Mip-NeRF 360 [Barron et al., 2022], Tanks & Temples [Knapitsch et al., 2017]) without staging scenes for spatially varying illumination.

We evaluate local light adaptation across multiple settings. First, we use the Laval Indoor Spatially Varying HDR dataset [Garon et al., 2019], widely used for spatially varying light estimation. Since Laval is not built for compositing, it provides no ground-truth composites; we therefore evaluate via a user study and a reference-free lighting-consistency measure. Second, for objective evaluation with ground truth, we collect LOCO-bench, an indoor benchmark of 760 physically captured composites in which each object is placed at multiple locations within a scene under spatially varying illumination. Third, we evaluate object compositing on the ORIDa benchmark [Kim et al., 2025] to verify that improved illumination adaptation preserves object identity. Across all settings, LOCO improves local illumination adaptation while preserving object identity.

Our main contributions are as follows:

1. We study object compositing under spatially varying illumination, a setting less explored in prior generative compositing work.
2. We present a physically grounded illumination-augmentation pipeline that turns real-world monocular videos into large-scale training data for local light-aware compositing.
3. We collect LOCO-bench, a benchmark of physically captured relit composites for objective evaluation under spatially varying illumination.

2 Related Work

Generative Object Compositing. Recent diffusion-based approaches have advanced object compositing by improving spatial alignment [Song et al., 2023, Chen et al., 2024a, Yu et al., 2025a, Tarrés et al., 2025, Winter et al., 2025], color harmonization [Wang et al., 2023, Hu et al., 2024], and plausibility cues such as shadows and reflections [Valença et al., 2023, Winter et al., 2024, Hong et al., 2022, Liu et al., 2024a, Tarrés et al., 2024, Wang et al., 2025b]. However, most methods implicitly assume spatially uniform illumination, offering no explicit supervision for local lighting adaptation. Several works introduce lighting variation in training data via synthetic 3D environments or game engines [Michel et al., 2024, Wang et al., 2025b], but these require costly scene authoring and operate with simplified lighting models. ObjectMover [Yu et al., 2025a] uses an environment lighting setup with offline illumination control, limiting its ability to model diverse localized effects. Other approaches address spatially varying illumination in specialized settings; SV Indoor Harmonization [Hu et al., 2024] targets human subjects using 3D scans but does not generalize to arbitrary objects. In contrast, our method targets general object compositing under spatially varying illumination through on-the-fly illumination augmentation from monocular videos, without staged capture or synthetic scene authoring.

Spatially-Varying Light Estimation. Another body of work explicitly estimates scene illumination. Early methods recover per-pixel illumination from single images [Zhu et al., 2022b, Li et al., 2022, 2020], while recent approaches combine illumination estimation with differentiable rendering to jointly recover geometry, BRDFs, and lighting [Zhu et al., 2022a, Wang et al., 2021, Peng et al., 2025, Wang et al., 2022, Bolduc et al., 2025]. These methods target physical scene reconstruction rather than object compositing, and typically rely on either multi-view inputs [Li et al., 2023, Srinivasan et al., 2020, Wu et al., 2023] or explicit 3D object assets [Zhu et al., 2021, Bai et al., 2023, Liang et al., 2024, Wang et al., 2022], limiting their use in practical editing scenarios. Rather than estimating lighting explicitly, our method learns to adapt object appearance to local illumination directly in image space, enabling compositing from a single background image and an object mask.

Intrinsic-Based Editing. A third paradigm tackles compositing through disentangled intrinsic spaces such as albedo–shading or reflectance representations [Zeng et al., 2024, Chen et al., 2024b, Zhang et al., 2024, Luo et al., 2024, Kocsis et al., 2024]. Compositing here follows a two-stage pipeline: inverse-rendering models estimate intrinsic representations, which are then rendered back to RGB via forward rendering [Careaga et al., 2023, Zhang et al., 2025, Liang et al., 2025]. However, single-image inverse and forward rendering are each ill-posed, and chaining them accumulates errors [Lyu et al., 2025, Liang et al., 2025]. These methods also assume geometric compatibility between object and scene, limiting applicability to constrained settings such as planar placement or object cloning. LOCO instead operates directly in RGB space without intrinsic decomposition, enabling end-to-end compositing under spatially varying illumination for arbitrary object poses and geometry.

3 Proposed Method: LOCO

As illustrated in Fig. 3, the goal of object compositing is, given a background image I_{bg}, an insertion mask M, and an object image I_{obj}, to synthesize a composite image I_{comp} whose object appearance matches the local illumination at the insertion region. We formulate this as a conditional diffusion problem with the rectified-flow objective [Liu et al., 2023, Lipman et al., 2023]. Our LOCO framework synthesizes illumination-augmented training data from monocular videos via physically based rendering (PBR), diversifying local lighting scenarios at training time.

3.1 The Illumination Augmentation Pipeline

Fig. 4 illustrates the proposed illumination augmentation pipeline, which performs *scene preparation* offline and applies *relighting* on the fly during augmentation.

Scene Preparation. Our pipeline applies to any monocular video of a static scene with sufficient viewpoint coverage.We use publicly available large-scale video datasets—uCO3D [Liu et al., 2025], DL3DV-10K [Ling et al., 2024], Mip-NeRF 360 [Barron et al., 2022], and Tanks & Temples [Knapitsch et al., 2017]—as primary data sources, supplemented by casually captured videos of everyday scenes. Foreground objects are segmented using Detic [Zhou et al., 2022] and SAM2 [Ravi et al., 2025], then removed with RoRem [Li et al., 2025] to obtain clean backgrounds. For re-

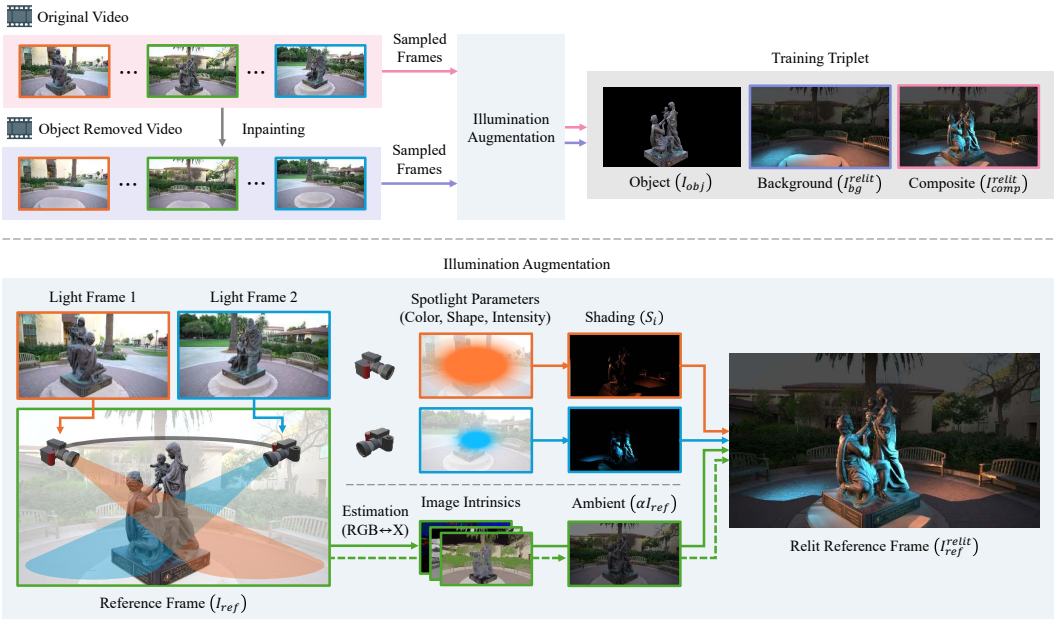


Figure 4: The LOCO pipeline. (Top) From a monocular video, we generate training triplets via illumination augmentation. (Bottom) Frames from the same video are treated as light views and instantiated as virtual spotlights with randomized color, shape, and intensity. Using estimated scene geometry and intrinsics, each light view projects spatially varying illumination onto a reference view via physically based relighting with complex lighting effects. This process enables efficient on-the-fly illumination augmentation.

lighting, we estimate and cache approximate scene geometry and reflectance parameters, including camera poses [Schönberger and Frahm, 2016], depth [Bochkovskii et al., 2025, Leroy et al., 2024], BRDF [Zeng et al., 2024], and original scene illumination [Li et al., 2020, Phongthawee et al., 2024]. These quantities are computed once offline and reused throughout the augmentation. The filtering pipeline for these estimates is detailed in Appendix A.9.

Each processed video yields training triplets $(I_{\text{bg}}, I_{\text{obj}}, I_{\text{comp}})$, where I_{comp} is a video frame, I_{obj} denotes the foreground segmented from I_{comp}, and I_{bg} is the object-removed background of the same frame. We uniformly sample up to 100 frames per scene, filter low-quality objects, and obtain ∼24K relightable objects, each observed from ∼28 viewpoints, yielding ∼671K unaugmented training triplets. At training time, we apply on-the-fly relighting and the permutation process described below.

Relighting. We treat illumination as a dynamically resampled augmentation variable applied to both I_{bg} and I_{comp}, ensuring that the input background and the supervision target share the same lighting configuration. To induce structured variation, we reinterpret other frames from the same video as pose-anchored *virtual spotlights*. Given reference frame I_{ref}, we sample N frames whose camera poses define candidate light positions, so that camera motion naturally yields diverse lighting directions. Since these poses lie in free space, face the scene, and are registered to the geometry, they enable occlusion-aware shading and shadows without manual light placement. We randomly perturb spotlight parameters such as beam shape, color, and intensity to produce diverse placement-dependent illumination patterns. Examples of relit realizations from a single reference frame are shown in Appendix Fig. 10. Parameter ranges and sampling details are provided in Appendix A.1.

To render illumination consistently with scene geometry, we employ rasterization-based shadow mapping. Given the reference depth map D_{ref}, each pixel x is back-projected to 3D and reprojected into the i-th light frame:

$$p(x) = o + D_{\text{ref}}(x)\,\omega_o(x), \quad \tilde{x}_i = K_i[R_i \mid t_i]\,[p(x), 1]^\top, \tag{1}$$

where o is the reference camera center, ω_o the viewing direction, and (K_i, R_i, t_i) the light-frame camera parameters. Visibility is determined via depth comparison, $V_i(x) = \mathbf{1}(D_i(\tilde{x}_i) \geq \|p(x) - o_i\|_2)$, enabling occlusion-aware shadow casting.

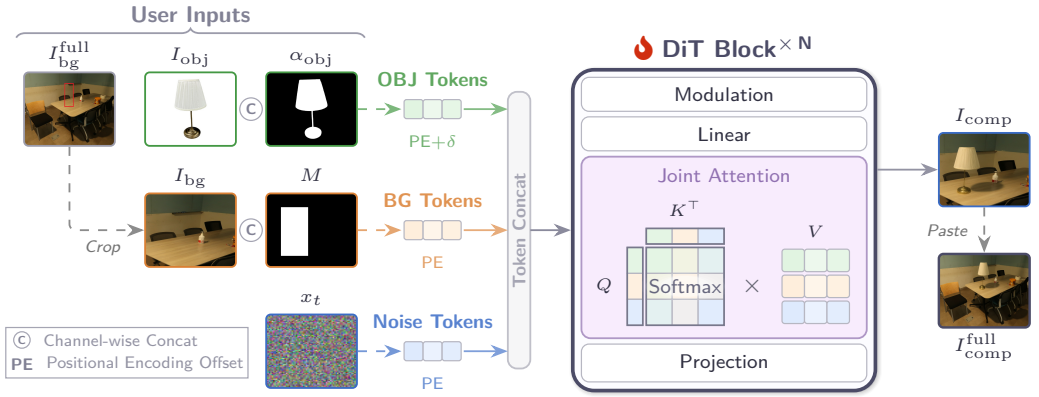


Figure 5: Conditioning layout. The object and the background crop, each channel-concatenated with its mask, are tokenized and processed jointly with the noisy latent x_t. α_{obj}: object alpha matte; M: insertion mask; PE: positional encoding, offset by δ on object tokens.

For each sampled light, we compute the incident radiance and resulting shading:

$$L_i(x, \omega_i) = \lambda_i \, V_i(x) \, M_i(\tilde{x}_i) \, \phi(x), \quad S_i(x) = L_i(x, \omega_i) \, R(x, \omega_o, \omega_i), \tag{2}$$

where λ_i controls intensity and color, M_i denotes the beam shape mask, $\phi(x)$ models distance falloff, and R is the simplified Disney BRDF [Burley and Studios, 2012].

Following prior work [Liang et al., 2025, 2024, Wang et al., 2022], we adopt a ratio-based illumination transfer, where each light contributes a multiplicative shading ratio to the reference image. The final relit image is obtained by summing these per-light contributions, with a small ambient term to prevent excessive darkening:

$$I_{\text{ref}}^{\text{relit}}(x) = \sum_{i=1}^{N} E_i(x) \, I_{\text{ref}}(x) + \alpha I_{\text{ref}}(x), \quad \text{where } E_i(x) = \frac{S_i(x)}{S_{\text{orig}}(x)}. \tag{3}$$

Here, $S_{\text{orig}}(x)$ denotes the shading under the original scene illumination, and α is an ambient strength. S_{orig} is independent of the sampled light configurations and therefore precomputed offline. Relighting requires only rasterization (depth reprojection and per-pixel shading), adding just 1.5% to the training step time (Appendix A.2).

Light and view batch permutation. Within each mini-batch, we recombine objects and backgrounds across light and view dimensions. For each scene, we sample V reference frames, each producing an original and a relit $(I_{\text{bg}}, I_{\text{comp}})$ pair, yielding $2V$ in total. Training triplets are formed by pairing one of these pairs with a foreground I_{obj} segmented from any of the $2V$ composites, allowing the objects and backgrounds to come from different views or lighting conditions. This expands the per-scene triplet count from V to $(2V)^2$; Fig. 6 shows a resulting triplet, where the object comes from a different view and lighting than the background. Such mismatched lighting discourages the model from simply copying the object's input shading. Instead, the background provides local lighting cues, while the corresponding relit composite serves as supervision for adapting the object's appearance to those conditions.

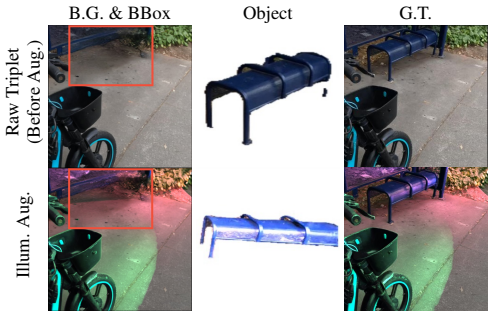


Figure 6: Training triplet example. The raw triplet shown (top) lacks spatially varying illumination and provides little supervision for adapting the object's appearance to the local lighting at the insertion location. Illumination augmentation (bottom) relights the scene under a sampled spotlight and pairs the relit background with an object from a different view and lighting.

Conditioning and training objective. We condition the diffusion model f_θ on $c = \{I_{\text{obj}}, I_{\text{bg}}, M\}$: the object image, a background crop around the insertion region, and their associated masks. Following prior compositing work [Chen et al., 2024a, Song et al., 2026], we define I_{bg} as a square crop

enclosing the insertion bbox, with side length proportional to the bbox size. Each image is encoded by a VAE [Kingma and Welling, 2014]; the corresponding mask is bilinearly resized to the latent grid, concatenated channel-wise with the latent, and projected to DiT token dimensions. All tokens are concatenated into a single sequence and processed by joint attention in the DiT backbone. To preserve spatial alignment between x_t and the background while keeping the object distinguishable, we share positional embeddings between x_t and the background and offset the object by a fixed δ, following OmniControl [Tan et al., 2025]. The model is supervised by the rectified-flow velocity loss

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{t,\epsilon}\left[\|f_\theta(x_t, c, t) - v\|_2^2\right], \tag{4}$$

where $x_t = (1-t)\, x_0 + t\, \epsilon$, $v = \epsilon - x_0$ is the velocity target, and x_0, ϵ, and t denote the ground-truth latent, Gaussian noise, and the time variable, respectively.

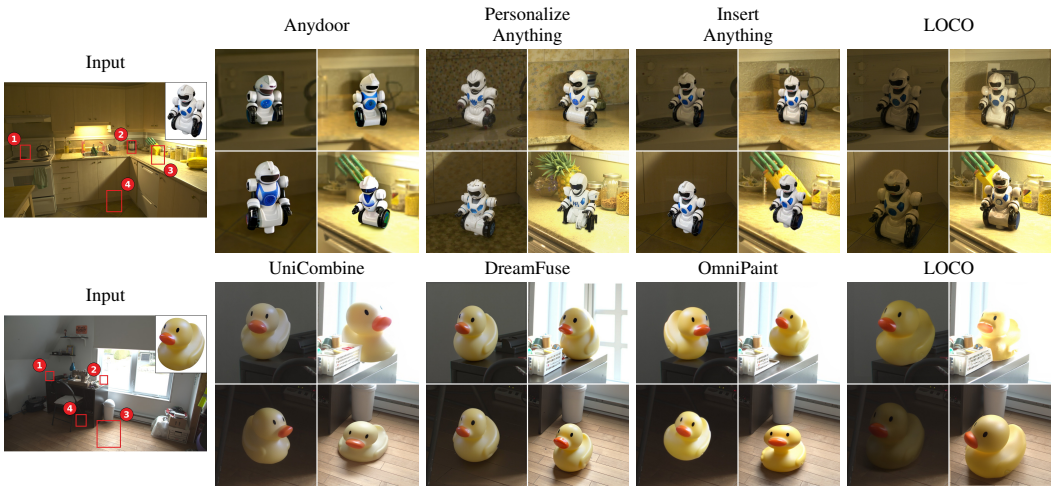
4 Experiments

4.1 Experimental Setup

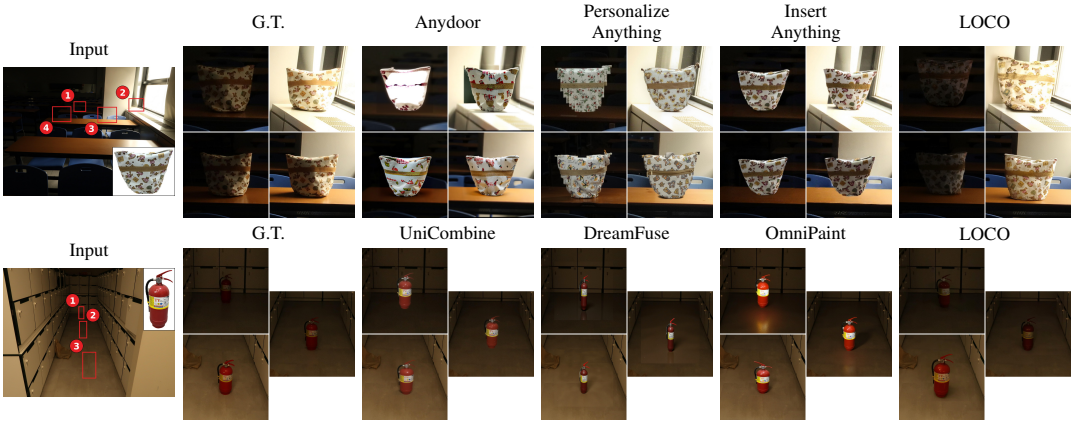
Benchmarks. We evaluate LOCO across multiple settings, spanning a conventional compositing benchmark, a light-estimation benchmark, and our newly collected relit-composite benchmark. (1) *Conventional object compositing*: The ORIDa evaluation set [Kim et al., 2025] provides real-world object–background pairs with ground-truth composites. As these scenes lack spatially varying illumination, they serve to evaluate general compositing capabilities such as identity preservation. (2) *Scenes with spatially varying illumination*: We insert 30 DreamBooth objects [Ruiz et al., 2023] into 20 scenes from Laval Indoor SV HDR dataset [Garon et al., 2019] at 3–5 positions per scene corresponding to the dataset's mirror-ball capture points, yielding 2,280 (object, background) pairs. Since ground-truth composites are unavailable, we evaluate via a user study and measure local lighting agreement before and after compositing. (3) *LOCO-bench*: We introduce LOCO-bench, a benchmark of 760 triplets (object image, background, ground-truth composite) under controlled spatially varying illumination, collected from 62 objects across 49 scenes. Backgrounds and ground-truth composites are physically captured from fixed viewpoints, while object references are photographed under different lighting conditions. Each object is placed at multiple locations per scene (up to 8, median 3), enabling direct evaluation of location-dependent appearance adaptation; construction details are in Appendix A.6.

Metrics. For identity preservation on ORIDa, we report CLIP [Radford et al., 2021] and DINO [Siméoni et al., 2025] similarity to the ground-truth composite, following Chen et al. [2024a], Wang et al. [2025a], Yu et al. [2025b]. On LOCO-bench, we use DreamSim distance [Fu et al., 2023] to the ground truth as the primary perceptual metric, capturing both identity and appearance (unlike DINO; see Appendix A.5). For Laval, which lacks ground-truth composites, we design a reference-free illumination consistency (IC) metric: the PSNR between estimated local lighting before and after compositing using a pretrained per-pixel lighting estimator [Li et al., 2020]. Computation details and validation of the metric are provided in Appendix A.5. On LOCO-bench, to compare more directly against the captured composite, we additionally measure, inside the insertion box, the light direction implied by the estimator's environment maps (L2 and angular error of the intensity-weighted mean direction), and the brightness, hue, and saturation of the bbox-mean color in HSV. Note that IC is reference-free, whereas these measurements compare against the ground truth. We also conduct a pairwise user study comparing LOCO against each baseline. Participants rate three aspects: *harmony* (whether the inserted object's shading looks natural in the scene), *identity* (whether the object's distinctive features are preserved), and *quality* (overall compositing quality). On Laval, judgments are made without a reference; on LOCO-bench, the ground-truth composite is shown alongside. Each participant evaluates 10 trials per dataset, with 38 valid participants on LOCO-bench and 33 on Laval. Further study details are provided in Appendix A.5 (Fig. 15).

Baselines. We compare against open-sourced state-of-the-art *generative compositing* methods: AnyDoor [Chen et al., 2024a], Personalize Anything [Feng et al., 2026], Insert-Anything [Song et al., 2026], Unicombine [Wang et al., 2025a], DreamFuse [Huang et al., 2025], OmniPaint [Yu et al., 2025b], and FreeCompose [Chen et al., 2024c]. For a thorough comparison, we evaluate each baseline with four background crop sizes ($1.5\times$, $3.0\times$, $4.5\times$, and $6.0\times$ the insertion mask) and report its best-performing configuration; per-scale results are in Appendix A.3. We provide additional comparisons with harmonization [Wang et al., 2023] and intrinsic-based editing methods [Zeng et al., 2024, Zhang et al., 2025, Liang et al., 2025] in Appendix A.3.



(a) Laval Indoor SV HDR



(b) LOCO-bench

Figure 7: LOCO adapts object appearance to local illumination at each insertion location. Surfaces darken at shadowed locations (e.g., Laval kitchen probe 4), while shading and color follow the local lighting (e.g., probes 2–3). In contrast, baselines show limited variation across locations; for example, Insert Anything mainly rescales overall brightness, with limited adaptation of shading to the local illumination. Cropped results follow the input numbering in clockwise order from the top-left. Ground-truth composites are available for LOCO-bench but not for Laval.

Implementation Details. For fair comparison with recent compositing baselines [Feng et al., 2026, Song et al., 2026, Wang et al., 2025a, Huang et al., 2025, Yu et al., 2025b], we adopt the Flux1-Dev backbone [Labs, 2024]. Training uses rank-16 LoRA [Hu et al., 2022] in bfloat16 with AdamW [Loshchilov and Hutter, 2019]. Starting from an AnyDoor-pretrained checkpoint as a generic generative-compositing prior, we fine-tune on a 4:1 mixture of our illumination-augmented data (§3.1) and the training sets of AnyDoor, AnyInsertion, ORIDa, and DreamFuse, which provide higher-resolution object crops complementary to our video-extracted data. During training, the background crop ranges from the bounding box to the full image, and is fixed to $1.5\times$ at inference. For each scene, we sample two reference frames and form 16 training triplets via batch permutation described in §3.1. Additional implementation details are provided in Appendix A.1.

4.2 Comparisons with Existing Methods

Appearance Adaptation. Fig. 7 shows LOCO adapting object appearance to local illumination across insertion locations, with consistent changes in brightness, shading, and color. Additional qualitative results and estimated irradiance maps [Zeng et al., 2024] are provided in Appendix (Fig. 12, 13). Quantitatively, LOCO achieves the lowest DreamSim distance on LOCO-bench (0.265 vs. Insert Anything 0.312, the strongest baseline; Tab. 1). In user studies shown on the right side of Tab. 1, our

Table 1: (Left) Quantitative comparison on scenes with spatially varying illumination. DreamSim [Fu et al., 2023] measures perceptual distance to ground-truth composites in LOCO-bench; IC is a reference-free metric (PSNR between estimated local lighting before and after compositing) used additionally on Laval [Garon et al., 2019], which lacks ground-truth composites. (Right) Pairwise user study, LOCO-win / tie / baseline-win (%); larger of W/L in bold. No-Ref denotes no-reference realism ranking on Laval, and Ref denotes reference-based comparison on LOCO-bench with ground-truth composites.

	Laval	LOCO-bench		Laval (No-Ref)			LOCO-bench (Ref)		
	IC↑	IC↑	DreamSim↓	Harmony	Identity	Quality	Harmony	Identity	Quality
AnyDoor	10.25	7.88	0.461	**96**/02/02	**76**/15/09	**91**/02/07	**88**/06/06	**88**/06/06	**92**/02/06
Personalize Anything	11.28	9.09	0.419	**100**/00/00	**61**/23/16	**96**/04/00	**85**/09/06	**74**/11/15	**91**/04/05
Insert Anything	12.12	11.10	0.312	**94**/06/00	**37**/33/30	**98**/00/02	**65**/14/21	**43**/14/**43**	**67**/14/19
UniCombine	13.17	11.80	0.342	**90**/08/02	**63**/25/12	**79**/13/08	**75**/06/19	**58**/09/33	**67**/08/25
DreamFuse	13.27	11.51	0.335	**82**/13/05	**64**/16/20	**71**/07/22	**70**/09/21	**68**/21/11	**79**/12/09
OmniPaint	11.92	9.54	0.349	**94**/02/04	**57**/29/14	**90**/02/08	**72**/11/17	**60**/13/27	**72**/13/15
FreeCompose	11.37	9.15	0.393	**100**/00/00	**84**/07/09	**95**/05/00	**91**/09/00	**74**/13/13	**93**/05/02
LOCO	**15.04**	**13.99**	**0.265**	–	–	–	–	–	–

Table 2: Illumination and appearance accuracy on LOCO-bench.

	Light direction		Color (HSV)		
	L2↓	Angle (°)↓	Hue (°)↓	Bright.↓	Sat.↓
AnyDoor	0.191	34.29	36.96	0.177	0.253
Personalize Anything	0.188	32.82	22.88	0.172	0.248
Insert Anything	0.147	26.36	14.33	0.107	0.175
UniCombine	0.150	26.55	13.06	0.103	0.178
DreamFuse	0.149	26.23	10.17	0.099	0.148
OmniPaint	0.171	30.30	14.48	0.150	0.194
FreeCompose	0.202	33.93	22.58	0.158	0.214
LOCO	**0.129**	**23.00**	**7.96**	**0.061**	**0.070**

Table 3: Compositing on the ORIDa [Kim et al., 2025] evaluation set.

	CLIP ↑	DINO ↑
Anydoor	86.6	79.4
Personalize Anything	74.4	52.2
Insert Anything	**90.9**	**92.1**
Unicombine	87.9	86.3
DreamFuse	83.1	67.7
OmniPaint	89.4	87.2
FreeCompose	81.0	78.2
LOCO	90.8	89.3

model is preferred over all baselines in harmony and overall quality. The reference-free IC metric is consistent: LOCO improves over the strongest baseline by 1.77 on Laval (vs. DreamFuse, 13.27) and 2.19 on LOCO-bench (vs. UniCombine, 11.80). Direct comparison with ground-truth composites (Tab. 2) further validates LOCO's adaptation to local illumination, reducing errors by 12% in light direction, 38% in brightness, and 22% in hue relative to the best-performing baselines. Additional comparisons show that LOCO also outperforms harmonization and intrinsic-based editing methods across IC, DreamSim, and ground-truth-based illumination and appearance metrics (Appendix Tabs. 9, 12 and Fig. 11).

Identity Preservation. On ORIDa, LOCO nearly matches Insert Anything in CLIP but trails it in DINO (Tab. 3). This aligns with the benchmark and metric: ORIDa lacks spatially varying lighting, and DINO emphasizes shape and texture similarity, favoring predictions that preserve the input appearance (Sec. A.5). As shown in Fig. 7, Insert Anything often retains the input object's original shading, mainly adjusting overall brightness rather than re-lighting the object (e.g., Laval and LOCO-bench probe 3). This yields strong identity scores on ORIDa but lower illumination consistency on Laval (IC 12.12 vs. 15.04; Tab. 1). In the user study, identity preferences between LOCO and Insert Anything are comparable (Laval: 37/33/30; LOCO-bench: 43/14/43; Tab. 1). Overall, LOCO maintains competitive identity preservation while better adapting object appearance to local lighting.

4.3 Ablation Study

We isolate the effect of illumination augmentation by varying the training data along two axes (Tab. 4): (i) fine-tuning on our raw triplets, with and without illumination augmentation; (ii) fine-tuning on prior compositing data, with and without our illumination-augmented triplets. On raw triplets, augmentation lowers LOCO-bench DreamSim from 0.372 to 0.308. On prior compositing data, DreamSim decreases from 0.309 to 0.265 when augmented triplets are added. This improvement is consistent with our data exhibiting a larger input–target appearance gap than prior datasets (Tab. 5: 0.504 vs. 0.17–0.30), providing stronger supervision for lighting adaptation. The reference-free Laval IC moves in the same direction across both axes (raw triplets $10.47 \rightarrow 14.51$ with augmentation;

Table 4: Effect of illumination augmentation, applied either to our raw triplets (top) or mixed with prior compositing data (bottom). Each row accumulatively adds one component to the previous configuration.

	Laval	LOCO-bench	
	IC↑	IC↑	DreamSim↓
Raw Triplets (No Aug.)	10.47	9.24	0.372
+ Illum. Aug.	14.51	13.52	0.308
+ Prior Comp. Data → **LOCO**	15.04	13.99	0.265
Prior Comp. Data	13.26	11.83	0.309
+ Our Illum. Aug. Data → **LOCO**	15.04	13.99	0.265

Table 5: Input–target appearance gap measured by DreamSim distance between the input object and the ground-truth composite. Larger values indicate stronger supervision for appearance adaptation.

Dataset	Appearance gap ↑
ORIDa	0.219
AnyInsertion	0.173
DreamFuse	0.299
LOCO	**0.504**

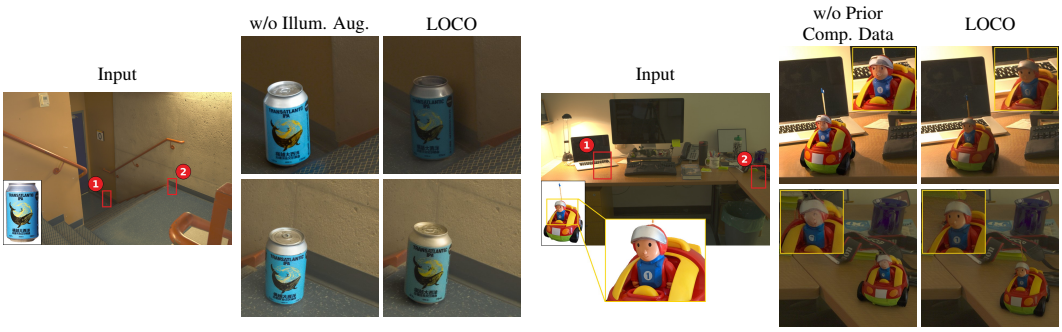


Figure 8: Qualitative ablation of illumination augmentation and prior compositing data. (Left) Illumination augmentation enables placement-dependent shading: without it, the inserted object remains brighter than the dim lighting at the insertion location and is inconsistent with the surrounding illumination gradients. (Right) Mixing prior compositing data better preserves details of the reference object (e.g., facial features and surface textures) that are less preserved with augmented triplets alone.

prior data $13.26 \rightarrow 15.04$). Taken together, these results suggest complementary roles for the two data sources: illumination augmentation primarily improves adaptation to local lighting, while prior compositing data help preserve object identity and fine details. Fig. 8 provides qualitative evidence for this distinction. Without illumination augmentation, the can retains similar shading across insertion locations, appearing overly bright in the dim region despite the surrounding illumination gradient. In contrast, training without prior compositing data still adapts object appearance to local lighting but less faithfully preserves fine details of the toy figure, including its eyes, nose, helmet, and the number on its shirt.

5 Conclusion

We presented LOCO, a framework for local light-aware object compositing. Treating monocular video frames as pose-anchored virtual spotlights can turn publicly available videos into illumination-augmented training data, equipping the compositing model with location-dependent shading. By varying illumination during training, our approach provides supervision for adapting object appearance to local lighting without requiring physically staged captures for each lighting configuration. Experiments on LOCO-bench with ground-truth relit composites, Laval Indoor SV HDR, and ORIDa show that LOCO outperforms prior methods in illumination adaptation while maintaining object identity. We will release the augmentation pipeline and LOCO-bench to support further research on local light-aware compositing.

Limitations: Our relighting pipeline relies on estimated scene geometry and reflectance, which can introduce shading artifacts for thin structures and highly specular surfaces. The Disney BRDF models surface reflectance but not transmission, leaving transparent and translucent materials outside the scope of the augmentation. Outdoor scenes pose a separate challenge, where a distant sun produces near-parallel illumination rather than the localized falloff modeled by our virtual spotlights. These limitations are left for future work.

Acknowledgments and Disclosure of Funding

This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT) (No. RS-2022-II220156, Fundamental research on continual meta-learning for quality enhancement of casual videos and their 3D metaverse transformation; No. RS-2026-25524173, Ultra-Long-Term Hierarchical Memory and Reasoning Architecture for Next-Generation Omnimodal Agents; No. RS-2025-25442338, AI Star Fellowship Support Program (Seoul National Univ.); No. RS-2025-02263841, Development of a Real-time Multimodal Framework for Comprehensive Deepfake Detection Incorporating Common Sense Error Analysis), by National IT Industry Promotion Agency (NIPA) grant funded by the Korea government (MSIT) (No. RS-2026-25621677, Physical Intelligence Action Model and Integrated Framework Technology Development), and by the "Advanced GPU Utilization Support Program" funded by the Government of the Republic of Korea (Ministry of Science and ICT).

References

Jiayang Bai, Zhen He, Shan Yang, Jie Guo, Zhenyu Chen, Yan Zhang, and Yanwen Guo. Local-to-global panorama inpainting for locale-aware indoor lighting prediction. In *TVCG*, 2023.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *CVPR*, 2022.

Black Forest Labs. FLUX.2 [klein]: Towards interactive visual intelligence. `https://bfl.ai/blog/flux2-klein-towards-interactive-visual-intelligence`, 2026.

Aleksei Bochkovskii, Amaël Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R. Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. In *ICLR*, 2025.

Christophe Bolduc, Yannick Hold-Geoffroy, Zhixin Shu, and Jean-François Lalonde. Gaslight: Gaussian splats for spatially-varying lighting in hdr. In *ICCV*, 2025.

Brent Burley and Walt Disney Animation Studios. Physically-based shading at disney. In *SIGGRAPH*, 2012.

Chris Careaga, S. Mahdi H. Miangoleh, and Yağız Aksoy. Intrinsic harmonization for illumination-aware compositing. In *SIGGRAPH ASIA Conference Proceedings*, 2023.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. Sam 3: Segment anything with concepts. In *ICLR*, 2025.

Xi Chen, Lianghua Huang, Yu Liu, Yujun Shen, Deli Zhao, and Hengshuang Zhao. Anydoor: Zero-shot object-level image customization. In *CVPR*, 2024a.

Xi Chen, Sida Peng, Dongchen Yang, Yuan Liu, Bowen Pan, Chengfei Lv, and Xiaowei Zhou. Intrinsicanything: Learning diffusion priors for inverse rendering under unknown illumination. In *ECCV*, 2024b.

Zhekai Chen, Wen Wang, Zhen Yang, Zeqing Yuan, Hao Chen, and Chunhua Shen. Freecompose: Generic zero-shot image composition with diffusion prior. In *ECCV*, 2024c.

Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. URL `http://www.blender.org`.

Mohammad Reza Karimi Dastjerdi, Jonathan Eisenmann, Yannick Hold-Geoffroy, and Jean-François Lalonde. Everlight: Indoor-outdoor editable HDR lighting estimation. In *ICCV*, 2023.

Epic Games. *Unreal Engine*, 2024. URL `https://www.unrealengine.com`.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *ICML*, 2024.

Haoran Feng, Zehuan Huang, Lin Li, Hairong Lv, and Lu Sheng. Personalize anything for free with diffusion transformer. In *AAAI*, 2026.

Stephanie Fu, Netanel Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. Dreamsim: Learning new dimensions of human visual similarity using synthetic data. In *NeurIPS*, 2023.

Mathieu Garon, Kalyan Sunkavalli, Sunil Hadap, Nathan Carr, and Jean-Francois Lalonde. Fast spatially-varying indoor lighting estimation. In *CVPR*, 2019.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Yannick Hold-Geoffroy, Kalyan Sunkavalli, Sunil Hadap, Emiliano Gambaretto, and Jean-François Lalonde. Deep outdoor illumination estimation. In *CVPR*, 2017.

Yan Hong, Li Niu, Jianfu Zhang, and Liqing Zhang. Shadow generation for composite image in real-world scenes. In *AAAI*, 2022.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022.

Zhongyun Hu, Jiahao Li, Xue Wang, and Qing Wang. Spatially-varying illumination-aware indoor harmonization. In *IJCV*, 2024.

Junjia Huang, Pengxiang Yan, Jiyang Liu, Jie Wu, Zhao Wang, Yitong Wang, Liang Lin, and Guanbin Li. Dreamfuse: Adaptive image fusion with diffusion transformer. In *ICCV*, 2025.

Jinwoo Kim, Sangmin Han, Jinho Jeong, Jiwoo Choi, Dongyoung Kim, and Seon Joo Kim. Orida: Object-centric real-world image composition dataset. In *CVPR*, 2025.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. In *ACM TOG*, 2017.

Peter Kocsis, Vincent Sitzmann, and Matthias Nießner. Intrinsic image diffusion for indoor single-view material estimation. In *CVPR*, 2024.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020.

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

Vincent Leroy, Yohann Cabon, and Jerome Revaud. Grounding image matching in 3d with mast3r. In *ECCV*, 2024.

Ruibin Li, Yang Tao, Guo Song, and Zhang Lei. Rorem: Training a robust object remover with human-in-the-loop. In *CVPR*, 2025.

Zhengqin Li, Mohammad Shafiei, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Inverse rendering for complex indoor scenes: Shape, spatially-varying lighting and svbrdf from a single image. In *CVPR*, 2020.

Zhengqin Li, Jia Shi, Sai Bi, Rui Zhu, Kalyan Sunkavalli, Miloš Hašan, Zexiang Xu, Ravi Ramamoorthi, and Manmohan Chandraker. Physically-based editing of indoor scene lighting from a single image. In *ECCV*, 2022.

Zhengqin Li, Li Yu, Mikhail Okunev, Manmohan Chandraker, and Zhao Dong. Spatiotemporally consistent hdr indoor lighting estimation. In *ACM TOG*, 2023.

Ruofan Liang, Zan Gojcic, Merlin Nimier-David, David Acuna, Nandita Vijaykumar, Sanja Fidler, and Zian Wang. Photorealistic object insertion with diffusion-guided inverse rendering. In *ECCV*, 2024.

Ruofan Liang, Zan Gojcic, Huan Ling, Jacob Munkberg, Jon Hasselgren, Zhi-Hao Lin, Jun Gao, Alexander Keller, Nandita Vijaykumar, Sanja Fidler, and Zian Wang. Diffusionrenderer: Neural inverse and forward rendering with video diffusion models. In *CVPR*, 2025.

Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *CVPR*, 2024.

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

Qingyang Liu, Junqi You, Jianting Wang, Xinhao Tao, Bo Zhang, and Li Niu. Shadow generation for composite image using diffusion model. In *CVPR*, 2024a.

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

Xingchen Liu, Piyush Tayal, Jianyuan Wang, Jesus Zarzar, Tom Monnier, Konstantinos Tertikas, Jiali Duan, Antoine Toisoul, Jason Y. Zhang, Natalia Neverova, Andrea Vedaldi, Roman Shapovalov, and David Novotny. Uncommon objects in 3d. In *CVPR*, 2025.

Zhiheng Liu, Ka Leong Cheng, Qiuyu Wang, Shuzhe Wang, Hao Ouyang, Bin Tan, Kai Zhu, Yujun Shen, Qifeng Chen, and Ping Luo. Depthlab: From partial to complete. *arXiv preprint arXiv:2412.18153*, 2024b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

Shilin Lu, Yanzhu Liu, and Adams Wai-Kin Kong. Tf-icon: Diffusion-based training-free cross-domain image composition. In *ICCV*, 2023.

Jundan Luo, Duygu Ceylan, Jae Shin Yoon, Nanxuan Zhao, Julien Philip, Anna Frühstück, Wenbin Li, Christian Richardt, and Tuanfeng Wang. Intrinsicdiffusion: Joint intrinsic layers from latent diffusion models. In *SIGGRAPH Conference Proceedings*, 2024.

Linjie Lyu, Valentin Deschaintre, Yannick Hold-Geoffroy, Miloš Hašan, Jae Shin Yoon, Thomas Leimkühler, Christian Theobalt, and Iliyan Georgiev. Intrinsicedit: Precise generative image manipulation in intrinsic space. In *ACM TOG*, 2025.

Oscar Michel, Anand Bhattad, Eli VanderBilt, Ranjay Krishna, Aniruddha Kembhavi, and Tanmay Gupta. Object 3dit: Language-guided 3d-aware image editing. In *NeurIPS*, 2024.

Weikun Peng, Sota Taira, Chris Careaga, and Yağız Aksoy. Interactive object insertion with differentiable rendering. In *SIGGRAPH Conference Proceedings*, 2025.

Pakkapon Phongthawee, Worameth Chinchuthakun, Nontaphat Sinsunthithet, Amit Raj, Varun Jampani, Pramook Khungurn, and Supasorn Suwajanakorn. Diffusionlight: Light probes for free by painting a chrome ball. In *CVPR*, 2024.

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 *ICML*, 2021.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. In *ICLR*, 2025.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *CVPR*, 2023.

Christophe Schlick. An inexpensive brdf model for physically-based rendering. In *Computer Graphics Forum (Eurographics)*, volume 13, pages 233–246, 1994.

Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. Dinov3, 2025. URL `https://arxiv.org/abs/2508.10104`.

Wensong Song, Hong Jiang, Zongxing Yang, Ruijie Quan, and Yi Yang. Insert anything: Image insertion via in-context editing in dit. In *AAAI*, 2026.

Yizhi Song, Zhifei Zhang, Zhe Lin, Scott Cohen, Brian Price, Jianming Zhang, Soo Ye Kim, and Daniel Aliaga. Objectstitch: Object compositing with diffusion model. In *CVPR*, 2023.

Pratul P. Srinivasan, Ben Mildenhall, Matthew Tancik, Jonathan T. Barron, Richard Tucker, and Noah Snavely. Lighthouse: Predicting lighting volumes for spatially-coherent illumination. In *CVPR*, 2020.

Zhenxiong Tan, Songhua Liu, Xingyi Yang, Qiaochu Xue, and Xinchao Wang. Ominicontrol: Minimal and universal control for diffusion transformer. In *ICCV*, 2025.

Gemma Canet Tarrés, Zhe Lin, Zhifei Zhang, Jianming Zhang, Yizhi Song, Dan Ruta, Andrew Gilbert, John Collomosse, and Soo Ye Kim. Thinking outside the bbox: Unconstrained generative object compositing. In *ECCV*, 2024.

Gemma Canet Tarrés, Zhe Lin, Zhifei Zhang, He Zhang, Andrew Gilbert, John Collomosse, and Soo Ye Kim. Multitwine: Multi-object compositing with text and layout control. In *CVPR*, 2025.

Yoad Tewel, Rinon Gal, Dvir Samuel, Yuval Atzmon, Lior Wolf, and Gal Chechik. Add-it: Training-free object insertion in images with pretrained diffusion models. In *ICLR*, 2025.

Lucas Valença, Jinsong Zhang, Michaël Gharbi, Yannick Hold-Geoffroy, and Jean-François Lalonde. Shadow harmonization for realistic compositing. In *SIGGRAPH Asia Conference Proceedings*, 2023.

Haoxuan Wang, Jinlong Peng, Qingdong He, Hao Yang, Ying Jin, Jiafu Wu, Xiaobin Hu, Yanjie Pan, Zhenye Gan, Mingmin Chi, et al. Unicombine: Unified multi-conditional combination with diffusion transformer. In *ICCV*, 2025a.

Ke Wang, Michaël Gharbi, He Zhang, Zhihao Xia, and Eli Shechtman. Semi-supervised parametric real-world image harmonization. In *CVPR*, 2023.

Tianyu Wang, Jianming Zhang, Haitian Zheng, Zhihong Ding, Scott Cohen, Zhe Lin, Wei Xiong, Chi-Wing Fu, Luis Figueroa, and Soo Ye Kim. Metashadow: Object-centered shadow detection, removal, and synthesis. In *CVPR*, 2025b.

Zian Wang, Jonah Philion, Sanja Fidler, and Jan Kautz. Learning indoor inverse rendering with 3d spatially-varying lighting. In *ICCV*, 2021.

Zian Wang, Wenzheng Chen, David Acuna, Jan Kautz, and Sanja Fidler. Neural light field estimation for street scenes with differentiable virtual object insertion. In *ECCV*, 2022.

Daniel Winter, Matan Cohen, Shlomi Fruchter, Yael Pritch, Alex Rav-Acha, and Yedid Hoshen. Objectdrop: Bootstrapping counterfactuals for photorealistic object removal and insertion. In *ECCV*, 2024.

Daniel Winter, Asaf Shul, Matan Cohen, Dana Berman, Yael Pritch, Alex Rav-Acha, and Yedid Hoshen. Objectmate: A recurrence prior for object insertion and subject-driven generation. In *ICCV*, 2025.

Liwen Wu, Rui Zhu, Mustafa B. Yaldiz, Yinhao Zhu, Hong Cai, Janarbek Matai, Fatih Porikli, Tzu-Mao Li, Manmohan Chandraker, and Ravi Ramamoorthi. Factorized inverse path tracing for efficient and accurate material-lighting estimation. In *ICCV*, 2023.

Xin Yu, Tianyu Wang, Soo Ye Kim, Paul Guerrero, Xi Chen, Qing Liu, Zhe Lin, and Xiaojuan Qi. Objectmover: Generative object movement with video prior. In *CVPR*, 2025a.

Yongsheng Yu, Ziyun Zeng, Haitian Zheng, and Jiebo Luo. Omnipaint: Mastering object-oriented editing via disentangled insertion-removal inpainting. In *ICCV*, 2025b.

Zheng Zeng, Valentin Deschaintre, Iliyan Georgiev, Yannick Hold-Geoffroy, Yiwei Hu, Fujun Luan, Ling-Qi Yan, and Miloš Hašan. Rgbx: Image decomposition and synthesis using material- and lighting-aware diffusion models. In *SIGGRAPH Conference Proceedings*, 2024.

Xiao Zhang, William Gao, Seemandhar Jain, Michael Maire, David Forsyth, and Anand Bhattad. Latent intrinsics emerge from training to relight. In *NeurIPS*, 2024.

Zitian Zhang, Frédéric Fortier-Chouinard, Mathieu Garon, Anand Bhattad, and Jean-François Lalonde. Zerocomp: Zero-shot object compositing from image intrinsics via diffusion. In *WACV*, 2025.

Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *ECCV*, 2022.

Jingsen Zhu, Fujun Luan, Yuchi Huo, Zihao Lin, Zhihua Zhong, Dianbing Xi, Rui Wang, Hujun Bao, Jiaxiang Zheng, and Rui Tang. Learning-based inverse rendering of complex indoor scenes with differentiable monte carlo raytracing. In *SIGGRAPH ASIA Conference Proceedings*, 2022a.

Rui Zhu, Zhengqin Li, Janarbek Matai, Fatih Porikli, and Manmohan Chandraker. Irisformer: Dense vision transformers for single-image inverse rendering in indoor scenes. In *CVPR*, 2022b.

Yongjie Zhu, Yinda Zhang, Si Li, and Boxin Shi. Spatially-varying outdoor lighting estimation from intrinsics. In *CVPR*, 2021.

A Supplementary Material

This supplementary material is organized into four groups.

Method
- §A.1 Implementation details.
- §A.2 Illumination augmentation analysis.

Experiments
- §A.3 Further baseline analysis.
- §A.4 Further analysis of LOCO.
- §A.5 Evaluation protocol details.

Data & benchmark
- §A.6 LOCO-bench construction.
- §A.7 Scene preparation details.
- §A.8 Training data statistics.
- §A.9 Training data filtering.

Discussion
- §A.10 Limitations.
- §A.11 Broader impact.
- §A.12 Image attribution.

A.1 Implementation Details

This subsection collects the design and training-loop details needed to reproduce LOCO: the diffusion backbone, the training schedule, inference settings, and the parameters that drive the illumination-augmentation pipeline.

Architecture. Our base model is Flux1-Dev [Labs, 2024], fine-tuned with rank-16 LoRA [Hu et al., 2022] in bfloat16. The conditioning encoding and rectified-flow objective are described in Sec. 3.1.

Training. We first pre-train on the AnyDoor [Chen et al., 2024a] training set for 10K iterations to obtain a generic generative-compositing prior. Starting from this checkpoint, we fine-tune for 20K iterations on a 4:1 mixture of our illumination-augmented data (Sec. 3.1) and the training sets of AnyDoor, AnyInsertion [Song et al., 2026], ORIDa [Kim et al., 2025], and DreamFuse [Huang et al., 2025]. Both phases use a per-GPU batch size of 32 across 4×B200 with gradient accumulation of 2, and AdamW [Loshchilov and Hutter, 2019] with learning rate 1×10^{-4}, β_1=0.9, β_2=0.999, and weight decay 1×10^{-4}. The full schedule takes approximately 4 days. Images are resized to 512×512. The background crop is a square enclosing the bbox with side length $s\cdot\max(w_{\text{bbox}}, h_{\text{bbox}})$; during training, $s\sim\mathcal{U}[1.1, 10]$, set to span the scene-to-bbox scales observed across the training data. During fine-tuning, we sample two reference frames per scene; with original and relit variants, this yields four (view, lighting) pairs, from which objects are cross-permuted to produce up to $4\times4=16$ training triplets per scene.

Inference. We run 28 Euler sampling steps with guidance scale 3.5. The latent is initialized from Gaussian noise and denoised over these steps. The background crop is taken at a fixed scale of $s=1.5$. To form the final composite, we decode the 512×512 crop prediction and blend its object region into the original background using the insertion mask, dilated and Gaussian-blurred at the boundary to smooth the transition. Pixels outside the blending region retain the original background. Higher-resolution refinement is discussed in Appendix A.10.

BRDF formulation. The reflectance R in Eq. 1 follows a simplified Disney BRDF [Burley and Studios, 2012] with diffuse and specular terms,

$$R(x,\omega_o,\omega_i)=\big(f_d(x)+f_s(x,\omega_o,\omega_i)\big)\,(\omega_i\cdot\mathbf{n}), \tag{5}$$

where $\mathbf{n}$ is the surface normal at x.

The diffuse term f_d is a Fresnel-modulated Lambertian lobe with surface albedo $\mathbf{a}$; the specular term f_s follows the Cook–Torrance microfacet model with half vector $\mathbf{h}=\text{normalize}(\omega_i+\omega_o)$ and roughness r,

$$f_d(x)=\frac{(1-F)\,\mathbf{a}(x)}{\pi},\qquad f_s(x,\omega_o,\omega_i)=\frac{D(\mathbf{h},\mathbf{n},r)\;F(\mathbf{h},\omega_o)\;G(\omega_i,\omega_o,\mathbf{n},r)}{4\,(\mathbf{n}\cdot\omega_i)\,(\mathbf{n}\cdot\omega_o)}. \tag{6}$$

The microfacet sub-terms use a spherical-Gaussian normal distribution D, the Schlick Fresnel approximation [Schlick, 1994] F with dielectric base reflectance $F_0 = 0.04$, and the Smith–Schlick–GGX geometry term G with $k = r^2/2$,

$$D(\mathbf{h}, \mathbf{n}, r) = \frac{1}{\pi\, r^4}\, \exp\Big(\frac{2}{r^4}\,\big((\mathbf{h}\cdot\mathbf{n}) - 1\big)\Big), \tag{7}$$

$$F(\mathbf{h}, \omega_o) = F_0 + (1 - F_0)\,(1 - \mathbf{h}\cdot\omega_o)^5, \tag{8}$$

$$G(\omega_i, \omega_o, \mathbf{n}, r) = \prod_{d\,\in\,\{i,o\}} \frac{\mathbf{n}\cdot\omega_d}{(\mathbf{n}\cdot\omega_d)(1-k)+k}. \tag{9}$$

All dot products are clamped to $[0, 1]$ for numerical stability.

Lighting parameters. Beam shapes are drawn from triangles, rectangles, and circles, then perturbed by random translation, rotation, and scaling, and combined with horizontal, vertical, or diagonal stripe masks of randomized thickness for varied patterns. Beam color is sampled in HSV: hue uniform in $[0, 1]$, saturation in $[0, 0.35]$, value 1.0, biasing the default sample toward near-white tones close to typical natural light; with probability 0.3 we instead draw a vivid color with saturation in $[0.5, 1.0]$ to cover artificial colored lighting (e.g., smart bulbs, neon signage). Since estimated reflectance and depth scale vary per scene, we set λ_i adaptively: per light view, we scale the beam intensity so the 90th-percentile relit pixel matches a fixed target, keeping apparent brightness comparable across scenes. For distance attenuation, we use $\phi(x) = 1/\big(1 + d^{\gamma}\big)$, where d is the source-to-surface distance and $\gamma > 0$ controls the rate of attenuation, ensuring a nonsingular falloff. The ambient term α is sampled in $[10^{-3},\, 0.2]$. From frames in which the object is visible, we sample up to 10 candidate light frames and apply them sequentially, accepting a candidate only if its lit region does not overlap with any previously accepted ones. We also reject lighting configurations that brighten the foreground object but barely illuminate the surrounding background; the rejection criteria and relighting quality checks are detailed in Appendix A.9.

A.2 Analysis of Illumination Augmentation

We analyze three properties of the augmented training data (spatially varying illumination, the input–target appearance gap, and lighting-direction coverage) and report the pipeline's computational footprint. Together these properties characterize the supervision signal: the augmentation needs to (a) produce spatially varying shading within each frame, so the model can infer local lighting cues at the insertion location, (b) widen the input–target appearance gap so the model cannot copy-and-paste, and (c) cover diverse lighting directions so the cues do not collapse to a single direction.

Spatially varying illumination. We check two properties: (i) the augmentation actually changes the image, and (ii) the change is spatially varying rather than a uniform brightness shift. For (i), pixel variance across multiple relit realizations of the same reference frame I_{ref} increases from 0.00 to 33.07 with augmentation (Tab. 6, top); Fig. 10 shows the same effect qualitatively: a single reference frame yields visibly different composites under different sampled lighting realizations. For (ii), spatially varying illumination produces bright and dark regions within the same frame, fattening the tails of the intensity histogram. We capture this with histogram contrast, the fraction of pixels beyond $\mu \pm 3\sigma$ computed per image. With augmentation, contrast rises from 0.0047 to 0.0151, whereas the global brightness or color jitter common in prior data pipelines [Song et al., 2023, Chen et al., 2024a, Kim et al., 2025] would yield 0 contrast change.

Input–target appearance gap. Training data teaches lighting adaptation only when the input object's appearance differs from that in the target composite; otherwise, the supervision provides little signal for adaptation. We measure this gap as DreamSim distance between the input object and the ground-truth composite, where larger values indicate stronger supervision. Our triplets reach 0.504, well above ORIDa [Kim et al., 2025], AnyInsertion [Song et al., 2026], and DreamFuse [Huang et al., 2025] (Tab. 6, bottom).

Lighting-direction coverage. Fig. 9 plots the angular distribution of our sampled light-view directions at the insertion region, derived from the camera poses of the reference and light frames. Frontal directions dominate, with additional coverage from side and above directions, while directions from below are rare, reflecting typical camera viewpoints in monocular videos. This diversity provides varied lighting cues and prevents collapse to a single direction.

	Pixel Var. ↑	Contrast↑
No Aug.	0.00	0.0047
LOCO	33.074	0.0151

	Appearance gap ↑
ORIDa	0.219
AnyInsertion	0.173
DreamFuse	0.299
LOCO	0.504

Table 6: (Top) Pixel variance and contrast across relit realizations. (Bottom) Input–target appearance gap (DreamSim); reproduced from Tab. 5. Higher values indicate stronger variation and a wider gap, respectively.

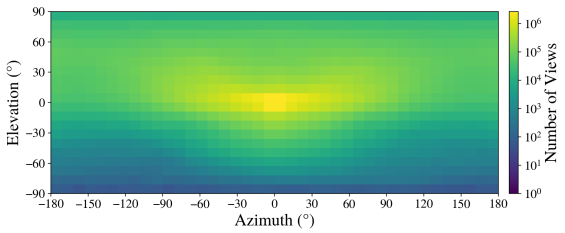


Figure 9: Sampled light-view directions, with frontal views dominant and additional coverage from lateral and oblique directions.

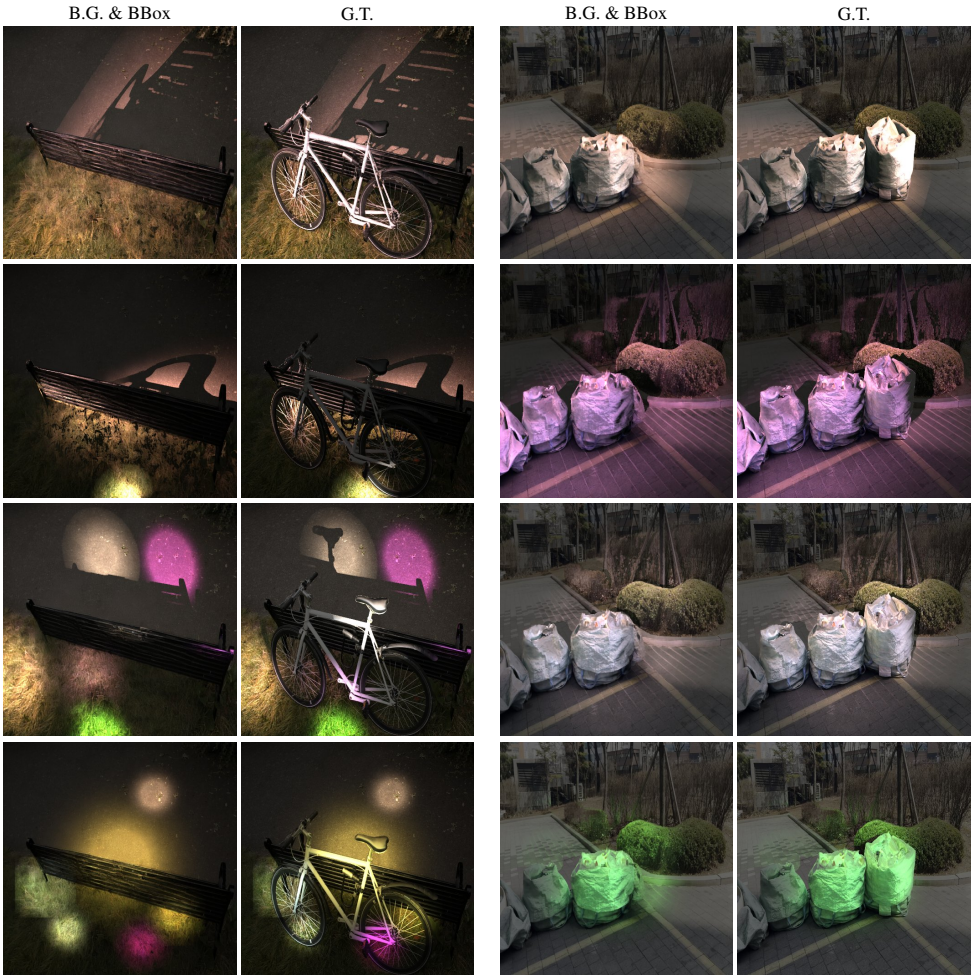


Figure 10: Appearance diversity from a single reference frame under sampled lighting. Each panel shows one reference frame; the four rows are independent realizations of $(I_{\text{bg}}, I_{\text{comp}})$ produced by the augmentation pipeline (Sec. 3.1), differing only in the sampled spotlight.

Computational cost. The augmentation pipeline runs on the fly within the training loop rather than as a separate offline stage. We measure wall-clock cost on a single NVIDIA B200: each iteration loads two reference frames with 5 sampled light views, yielding 16 training triplets (Flux1-Dev with rank-16 LoRA in bf16; 45 measured steps after 5 warmup). Per iteration, augmentation takes a median of 83 ms while the full step (augmentation, forward, backward, parameter update) takes 5.40 s; augmentation occupies approximately 1.5% of step time. Peak GPU memory growth during augmentation is 1.08 GB (median; 1.11 GB max), small relative to the model's training footprint. Under the full training batch (two scenes per GPU, 32 variants), costs scale near-linearly: median augmentation 168 ms and full step 10.74 s, with a similar $\sim 1.5\%$ ratio. This low overhead allows augmentation to run in the loop, exposing the model to freshly relit triplets at each step.

A.3 Further Baseline Analysis

This subsection extends the baseline coverage of the main paper by evaluating generative compositing baselines at different background-context crop scales and comparing against harmonization, intrinsic-based editing, and general-purpose multi-reference editing.

Effect of background context scale. Background crop scale affects methods that operate on local context. We evaluate the generative compositing baselines at four scales ($1.5\times$, $3.0\times$, $4.5\times$, and $6.0\times$ the bbox side) on the evaluation set of ORIDa (identity) and LOCO-bench (illumination). Identity is measured using CLIP ViT-L/14 [Radford et al., 2021] and DINOv3 ViT-H/16 [Siméoni et al., 2025]. Results are reported in Tab. 7 and Tab. 8; main-paper comparisons use each method's best-performing scale.

Table 7: Effect of background context scale on identity preservation (ORIDa). Each baseline is evaluated across four crop scales; the best-performing scale is used in the main paper.

	1.5×		3.0×		4.5×		6.0×	
	CLIP↑	DINO↑	CLIP↑	DINO↑	CLIP↑	DINO↑	CLIP↑	DINO↑
AnyDoor	86.6	79.4	86.0	74.6	86.1	74.5	86.0	74.5
Personalize Anything	74.4	52.2	71.7	34.5	69.9	28.7	69.4	27.6
Insert Anything	90.9	92.1	91.1	89.4	90.1	88.9	90.0	90.4
UniCombine	86.8	85.0	87.9	86.3	86.2	81.4	83.9	75.1
DreamFuse	83.1	67.7	79.2	57.4	76.3	46.7	76.6	46.6
OmniPaint	89.3	87.2	89.4	87.2	87.8	85.1	87.0	84.0
FreeCompose	81.0	78.2	79.7	71.3	80.1	71.8	80.0	72.2

Table 8: Effect of background context scale on appearance adaptation (LOCO-bench). Each baseline is evaluated across four crop scales; the best-performing scale is used in the main paper.

	1.5×	3.0×	4.5×	6.0×
	DreamSim↓	DreamSim↓	DreamSim↓	DreamSim↓
AnyDoor	0.477	0.461	0.476	0.501
Personalize Anything	0.419	0.461	0.489	0.513
Insert Anything	0.359	0.323	0.313	0.312
UniCombine	0.350	0.342	0.399	0.467
DreamFuse	0.335	0.351	0.377	0.394
OmniPaint	0.357	0.349	0.365	0.393
FreeCompose	0.393	0.410	0.430	0.448

Harmonization and intrinsic-based editing. To extend baseline coverage beyond the generative compositing methods in the main paper, we add two complementary paradigms: (1) *harmonization*: PIH [Wang et al., 2023], which transfers color and tone to match the surrounding scene; (2) *intrinsic-based editing*: RGBX [Zeng et al., 2024], ZeroComp [Zhang et al., 2025], and DiffusionRenderer [Liang et al., 2025], which decompose the image into intrinsics (e.g., albedo, shading) and re-render under target lighting. Both paradigms run on full images so they retain global lighting context. We use official weights and default hyperparameters for every baseline.

Tab. 9 reports IC and DreamSim [Fu et al., 2023] on LOCO-bench under the same protocol as the main paper, and Fig. 11 aligns the same insertion under each method. On IC, LOCO scores higher than all four baselines, exceeding the next-best ZeroComp by 0.29 PSNR; on DreamSim, LOCO (0.265) is lower than every baseline (0.351–0.554).

Table 9: Quantitative comparison with harmonization and intrinsic-based editing methods on LOCO-bench.

	IC↑	DreamSim↓
PIH	11.79	0.351
RGBX	13.13	0.554
ZeroComp	13.70	0.447
DiffusionRenderer	12.38	0.389
LOCO	13.99	0.265



Figure 11: PIH adjusts the overall color and tone of the inserted object but leaves shading and material cues unchanged. RGBX, ZeroComp, and DiffusionRenderer adjust brightness and shading through inverse-then-forward rendering, but do not explicitly model geometric placement on the supporting surface, so the inserted object often appears visually detached and lacks contact cues such as cast shadows.

General-purpose multi-reference editing. We evaluate FLUX-2-Klein-9B [Black Forest Labs, 2026] using two reference images: a background crop with the insertion bounding box marked in red, and a masked object reference. The model is instructed to insert the reference object at the marked location, adapt its brightness, shading, color, and cast shadow to the local lighting, remove the red marker, and preserve the rest of the scene. The full prompt is:

> Insert the object from the second image into the first image, exactly inside the region marked by the red box. Place it naturally on the supporting surface. Adapt the object's brightness, shading, color, and cast shadow to match the local lighting at that spot in the scene. Remove the red box marker. Do not change anything else in the scene.

FLUX-2-Klein-9B achieves a lower DreamSim distance than all specialized compositing baselines in Tab. 1 (0.293 vs. the best baseline's 0.312). LOCO achieves a lower DreamSim distance (0.265) and higher IC on both benchmarks (Tab. 10).

Table 10: Comparison with a general-purpose multi-reference editor.

Method	Laval IC↑	LOCO-bench IC↑	LOCO-bench DreamSim↓
FLUX-2-Klein-9B	12.48	11.05	0.293
LOCO	**15.04**	**13.99**	**0.265**

A.4 Further Analysis of LOCO

This subsection examines the effect of the video training corpus, followed by qualitative comparisons against the main-paper baselines, an estimated-irradiance analysis of shading consistency, and comparisons with captured ground truth on challenging LOCO-bench samples.

Contribution of Public Video Data. The augmentation pipeline can use ordinary monocular videos, ranging from a small set of casual captures to a large public corpus. Here we examine whether a small collection of casual captures provides effective training supervision and what is gained by expanding to a larger public video corpus, complementing the training-data ablation in Sec. 4.3.

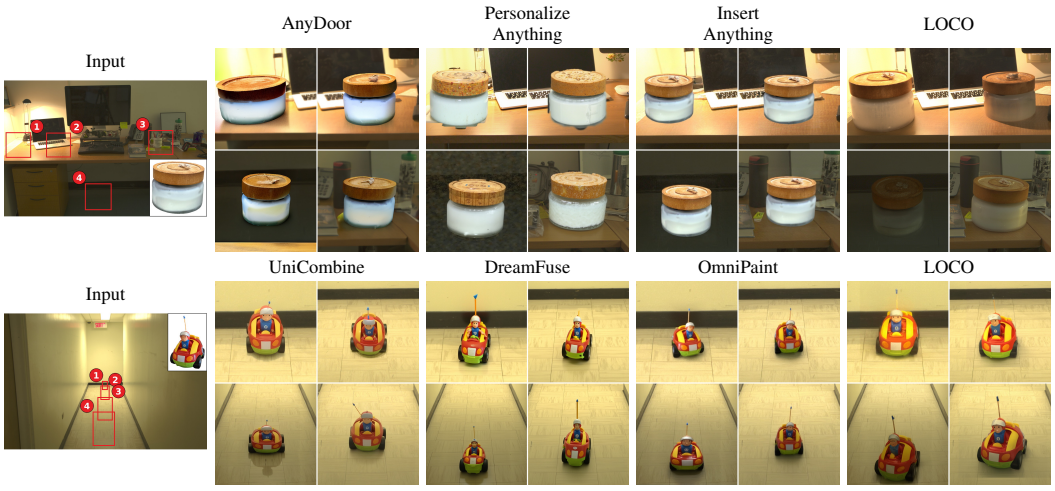
Tab. 11 compares three fine-tuning configurations: prior compositing data alone; those data combined with augmented triplets from our 51 casual captures; and the full training mixture, which additionally uses the public video datasets in Tab. 13. The casual-capture configuration uses no public videos for augmentation; the full configuration adds DL3DV, uCO3D, Mip-NeRF 360, and Tanks&Temples together. All configurations use the same initial checkpoint trained on the AnyDoor dataset and retain the same prior compositing datasets. Both video configurations use the same augmentation and light/view permutation pipeline. The casual captures are separate from LOCO-bench and contribute 12,433 of the 670,899 unaugmented training triplets (1.9%).

Training with prior compositing data alone gives results close to the best generative baseline scores listed in the table. With augmented triplets from only our 51 casual captures, the model surpasses these references on all three metrics: Laval IC increases from 13.26 to 14.65, LOCO-bench IC from 11.83 to 13.50, and LOCO-bench DreamSim improves from 0.309 to 0.288. Adding the public videos further improves these metrics to 15.04, 13.99, and 0.265, respectively. Most of the IC improvement is already achieved with the augmented casual captures, while adding public videos yields a further DreamSim reduction comparable to the initial improvement ($0.309 \rightarrow 0.288$ and $0.288 \rightarrow 0.265$, respectively). These results show that our pipeline provides effective illumination supervision from a small collection of captures, while a larger and more diverse video corpus further improves perceptual fidelity.

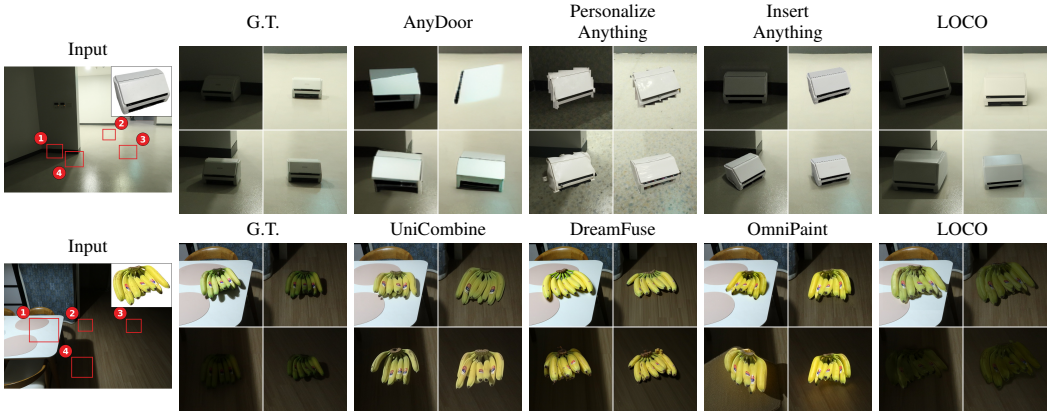
Table 11: Effect of the video collection used for augmented training triplets. All three training configurations start from the same checkpoint trained on the AnyDoor dataset and retain the prior compositing data from Tab. 4. The two video configurations use the same illumination augmentation and light/view permutation pipeline. The reference baseline row combines the per-column best generative compositing results from Tab. 1: DreamFuse for Laval IC, UniCombine for LOCO-bench IC, and Insert Anything for DreamSim.

	Laval	LOCO-bench	
Video collection used for augmentation	IC↑	IC↑	DreamSim↓
Best generative baseline (per column)	13.27	11.80	0.312
None (prior compositing data only)	13.26	11.83	0.309
Our 51 casual captures	14.65	13.50	0.288
Full corpus: casual captures and public videos (LOCO)	**15.04**	**13.99**	**0.265**

Additional composite samples. Fig. 12 extends the main-paper comparison (Fig. 7) with additional Laval and LOCO-bench scenes, where LOCO adapts object appearance to local illumination across insertion locations with consistent changes in brightness and cast shadow.



(a) Laval Indoor SV HDR



(b) LOCO-bench

Figure 12: LOCO adapts object appearance to local illumination at each insertion location. Surface brightness and cast shadow vary with placement. In contrast, baselines respond only weakly to the local light context: they fail to match the local brightness (Laval top, LOCO-bench bottom) or to produce a plausible cast shadow (Laval top, LOCO-bench top). Cropped results follow the input numbering in clockwise order from the top-left. Ground-truth composites are available for LOCO-bench but not for Laval.

Estimated irradiance. As an additional qualitative comparison on local lighting consistency, we estimate an irradiance map from each composite using RGBX [Zeng et al., 2024] and visualize it alongside the composite (Fig. 13; top: composite, bottom: estimated irradiance). For baselines, the irradiance on the inserted object often differs from its surroundings, with the boundary discontinuity becoming more pronounced as the composite's shading deviates further from the local lighting. In contrast, LOCO yields irradiance that aligns with the surroundings and transitions smoothly across the boundary, reflecting consistent adaptation to insertion location.

The estimator can be unreliable in regions with clipped highlights, where the recovered irradiance may reflect estimation artifacts rather than true illumination. For example, in the clock panel (second row), the desk surface around probe 1 contains strong highlights, and the estimates are noisy across all methods.

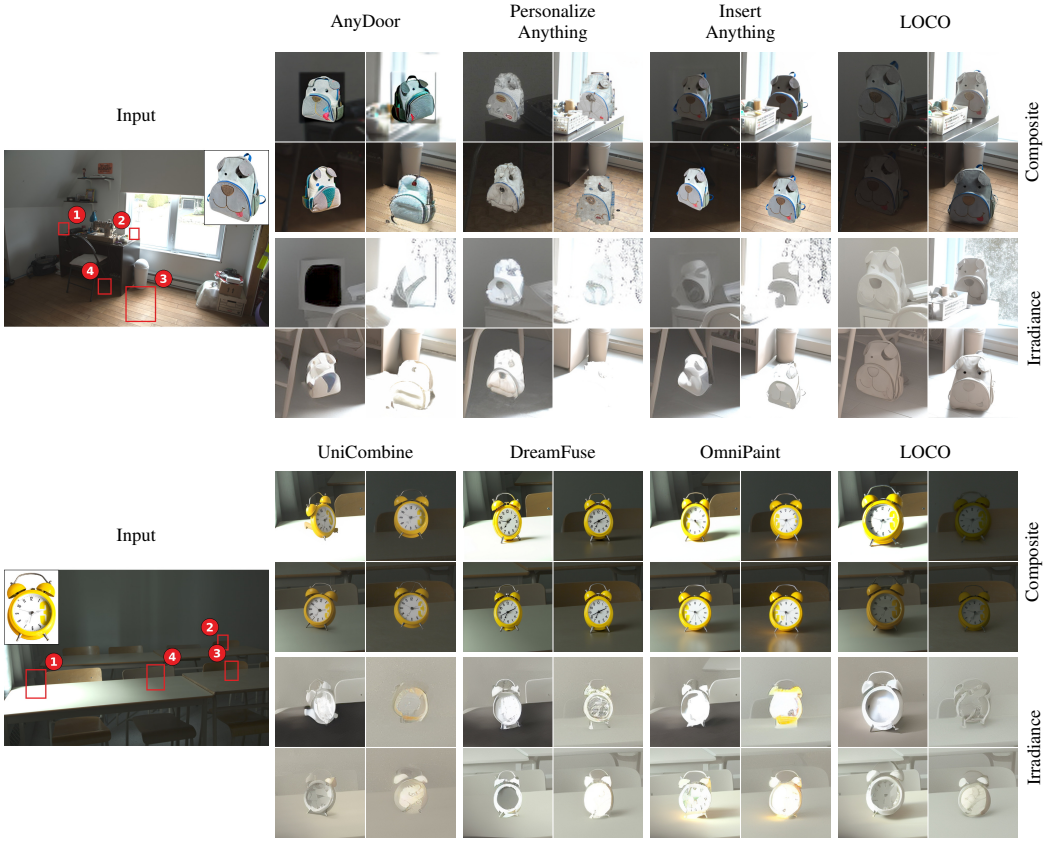


Figure 13: Estimated-irradiance comparison [Zeng et al., 2024] across generative compositing baselines. For baselines, the irradiance on the inserted object often differs from its surroundings, with the boundary discontinuity becoming more pronounced as the composite's shading deviates further from the local lighting. In contrast, LOCO yields irradiance that aligns with the surroundings and transitions smoothly across the boundary. For each input, the top row shows the composite output and the bottom row shows the estimated irradiance computed from that composite.

Comparison with ground truth on LOCO-bench. Fig. 14 compares LOCO with captured ground truth at two insertion locations per scene. The examples highlight remaining differences in fine shading, partially observed regions, and transmissive materials.

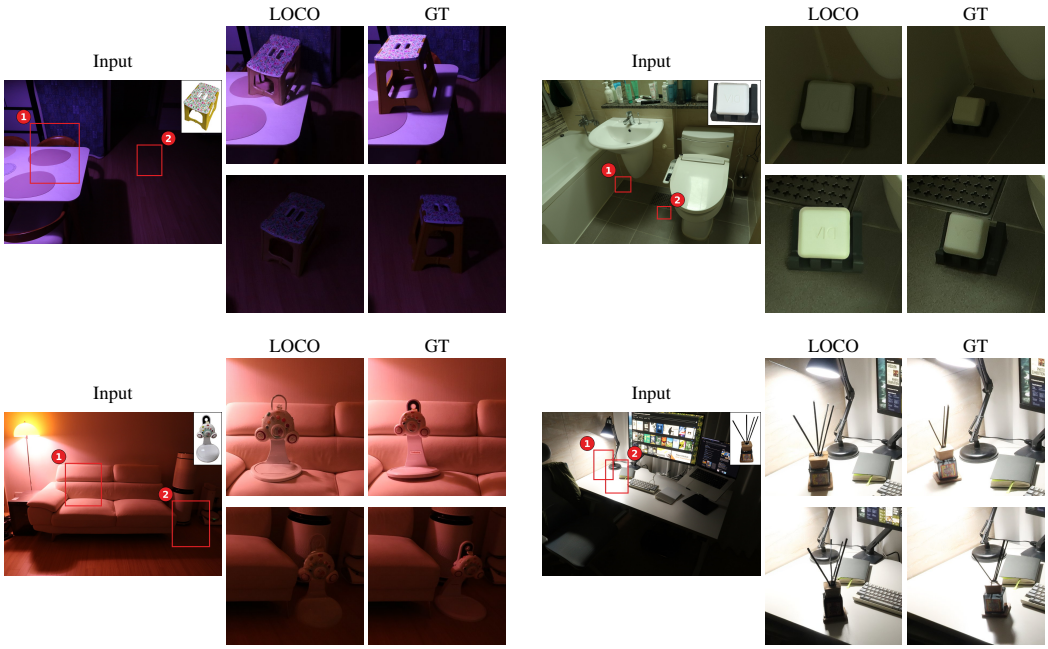


Figure 14: Challenging samples from LOCO-bench under spatially varying illumination. Each panel shows one object-scene example: the input composite on the left, and two (LOCO, ground truth) comparisons on the right, corresponding to two locations within the same scene. The examples include strong colored lighting, transmissive materials, and viewpoints where parts of the object are weakly observed in the input. While LOCO reproduces the dominant lighting effects, comparison with the ground truth reveals remaining differences in fine shading details, particularly on regions partially occluded in the input view, as well as transmissive effects.

A.5 Evaluation Protocol Details

This subsection documents four evaluation choices: how the user study was conducted, why we adopt DreamSim as the primary perceptual metric on LOCO-bench, the rationale and computation of the IC metric, and the direct comparison with ground-truth composites that complements IC on LOCO-bench.

User study. We run a pairwise user study comparing LOCO against each baseline. Each trial presents LOCO and a randomly sampled baseline on the same (object, scene) pair, with randomized left/right ordering. Above the candidates, we show the input object and the target background scene. The candidates contain the object inserted at multiple locations; for LOCO-bench, the ground-truth composite is additionally shown.

Participants then judge the two candidates on three criteria:

- *Harmony*: whether the inserted object's shading matches the surrounding illumination
- *Identity*: whether the object's shape, texture, and color are preserved
- *Quality*: overall compositing quality, including illumination plausibility and visual coherence

For each criterion the participant chooses one of *A* (left), *B* (right), or *Tie*. Laval Indoor SV HDR is evaluated without a reference, while LOCO-bench includes the ground-truth composite.

Protocol. The study is conducted through a custom web-based interface (Fig. 15). Each participant completes 10 trials per dataset. We collect 38 responses on LOCO-bench and 33 on Laval after excluding inattentive responses; we exclude a participant if they choose the same option (A, B, or Tie) across all three criteria in all 10 trials.

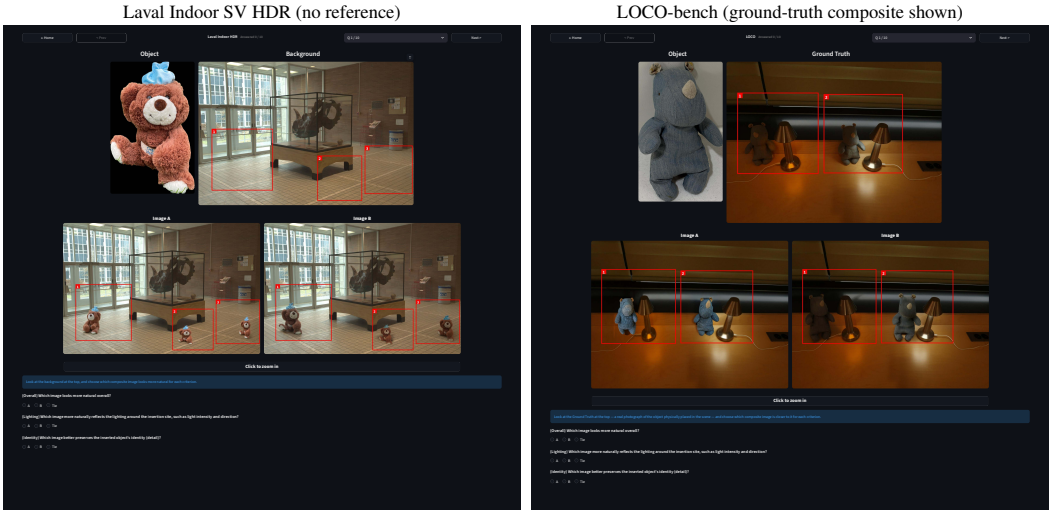


Figure 15: Web interface used for the user study. For each trial, two composites are shown side by side, with a reference (ground-truth composite) above on LOCO-bench and no reference on Laval Indoor SV HDR. Participants choose A / B / Tie for each of the three criteria.

Choice of perceptual metric. On LOCO-bench, the ground truth contains the object *relit* to match the local illumination, so a suitable metric should reflect alignment in appearance rather than identity alone. DINO produces features designed to be robust to style variations [Siméoni et al., 2025], placing greater emphasis on shape and texture similarity than on appearance differences such as shading. As a result, a *copy-and-paste* prediction (directly inserting the input object without relighting) can still achieve a high DINO score.

DreamSim [Fu et al., 2023] is a learned perceptual metric trained on human similarity judgments over mid-level attributes such as style, color, pose, and layout, making it more sensitive to appearance differences that include shading. Predictions whose shading better matches the ground truth tend to receive lower DreamSim distance, while copy-and-paste composites are penalized. Fig. 16 illustrates this on LOCO-bench samples: Insert Anything largely preserves the input object's appearance and

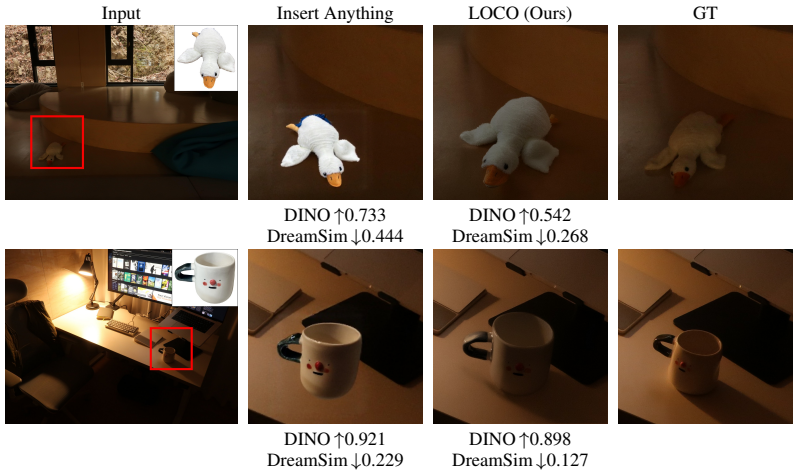


Figure 16: Sample-level disagreement between DINO and DreamSim on LOCO-bench. Insert Anything largely preserves the input object's appearance, so its prediction remains close in shape and texture and receives a higher DINO score than LOCO's relit output. DreamSim, which reflects perceptual differences in attributes such as color and shading, assigns a lower (better) distance to LOCO, consistent with visual similarity to the ground truth. Each score is computed between the prediction and the ground-truth crop in the same row.

scores higher on DINO than LOCO, whereas LOCO's relit prediction is closer to the ground truth under DreamSim. We therefore adopt DreamSim as the primary perceptual metric on LOCO-bench.

IC metric details. DreamSim requires a ground-truth composite as reference, which limits its applicability to in-the-wild scenes such as Laval Indoor SV HDR. For such settings, we additionally use a reference-free illumination consistency (IC) measure, which operates on an estimated lighting field and provides a signal largely independent of object identity.

The intuition behind IC is that the scene's light sources remain unchanged by insertion. If the inserted object's shading matches the local lighting, an illumination estimator's prediction at the insertion region should remain close to its pre-compositing prediction; mismatched shading may shift it. We use Li et al. [Li et al., 2020] for its per-pixel predictions.

Computation. For each pixel inside the insertion bounding box, the estimator outputs an 8×16 equirectangular environment map (decoded from 12 spherical Gaussians) oriented to the pixel's surface normal. To preserve angular detail under normal-based rotations, we reproject each per-pixel envmap onto a shared higher-resolution grid (64×128) in world coordinates and average over all pixels in the bounding box. IC is computed as the PSNR between this bbox-level envmap before and after compositing.

Validation with ground truth. LOCO-bench provides ground-truth composites, allowing a direct examination of the metric. All evaluated methods score below the ground truth (14.52), with scores ranging from 7.88 (AnyDoor) to 13.99 (LOCO), and UniCombine as the strongest baseline at 11.80. LOCO is 2.19 PSNR above the strongest baseline and 0.53 PSNR below the ground truth.

Relation to human judgments. We further examine how IC relates to human preferences by comparing each baseline's IC with its harmony preference rate in the user study (§A.5). The two tend to move in the same direction. On Laval (no-reference), the two highest-IC baselines, DreamFuse (13.27) and UniCombine (13.17), correspond to lower LOCO harmony win rates (82% and 90%), while the four lowest-IC baselines are at 94% or above (Spearman ρ=0.87). On LOCO-bench (reference-based), a similar but weaker trend is observed (Spearman ρ=0.64). These observations suggest that IC captures aspects of local-lighting plausibility reflected in human judgments, although it does not replace human evaluation.

A practical caveat is the limited availability of publicly available per-pixel illumination estimators. Li et al., the most recent we are aware of, outputs an 8×16 environment map per pixel. The IC definition is independent of the estimator; substituting a stronger per-pixel model would likely improve measurement fidelity without altering the underlying metric.

Illumination and appearance accuracy. IC compares the estimated lighting field before and after insertion, but it does not tell which aspect of the local illumination a method reproduces. To complement it, we compare each prediction directly with the captured ground-truth composite on all 760 LOCO-bench triplets, inside each insertion bounding box, on three quantities: the implied light direction, and the brightness and color of the inserted object.

Light direction. Extracting a single shadow axis is unreliable under overlapping indoor lights, and some methods change the object's orientation. We therefore compare the light direction implied by each prediction with that of the ground truth. From the per-pixel environment maps of the IC estimator, we compute the intensity-weighted mean direction over the bounding box and report (i) the L2 distance between the weighted mean directions and (ii) the angular error between their normalized directions. The unnormalized L2 distance down-weights diffuse regions, where direction is poorly defined.

Local brightness. We convert the mean RGB inside the bounding box to HSV and report the absolute difference in value, $|\Delta V|$, against the ground truth.

Color shift. From the same HSV values, we report the cyclic hue difference, $\min(|\Delta h|,\ 360^\circ - |\Delta h|)$, and the absolute saturation difference $|\Delta S|$. We use hue and saturation rather than a scalar color temperature, which is undefined for arbitrary RGB illumination.

Tab. 12 reports all five readouts for the generative compositing baselines of the main paper and for the harmonization and intrinsic-based methods of §A.3. LOCO attains the best score on every measure. Against the generative compositing baselines, it improves on the per-column best by 12% in light direction (both readouts), 22% in hue, 38% in brightness, and 53% in saturation. The strongest

baseline on the direction and brightness axes is PIH, a harmonization method, which LOCO leads by 12% and 28%. Fig. 13 provides the corresponding spatial evidence: the irradiance estimated on LOCO's composites follows the local brightness and shading variation of the scene.

Table 12: Illumination and appearance accuracy on LOCO-bench (760 triplets). Each quantity is measured inside the insertion bounding box against the captured ground-truth composite; lower is better. Light direction is the intensity-weighted mean direction of the per-pixel environment maps from the IC estimator, reported as an L2 distance and as an angular error. Brightness, hue, and saturation are differences of the bbox-mean color in HSV. The generative compositing rows are also shown in main paper Tab. 2.

	Light direction		Color (HSV)		
Method	L2↓	Angle (°)↓	Hue (°)↓	Brightness↓	Saturation↓
Generative compositing (main paper)					
AnyDoor	0.191	34.29	36.96	0.177	0.253
Personalize Anything	0.188	32.82	22.88	0.172	0.248
Insert Anything	0.147	26.36	14.33	0.107	0.175
UniCombine	0.150	26.55	13.06	0.103	0.178
DreamFuse	0.149	26.23	10.17	0.099	0.148
OmniPaint	0.171	30.30	14.48	0.150	0.194
FreeCompose	0.202	33.93	22.58	0.158	0.214
Harmonization and intrinsic-based editing (§A.3)					
PIH	0.146	26.21	18.34	0.085	0.145
RGBX	0.237	40.05	11.23	0.268	0.235
ZeroComp	0.178	31.52	15.16	0.090	0.126
DiffusionRenderer	0.149	26.40	20.24	0.119	0.114
LOCO	**0.129**	**23.00**	**7.96**	**0.061**	**0.070**

A.6 LOCO-bench Construction

Overview. LOCO-bench consists of 760 (object, scene, ground-truth composite) triplets over 62 objects and 49 scenes. For each scene, we first photograph the object under a lighting environment different from the target, then physically place it into the scene and capture the composite under the target lighting. This deliberate illumination mismatch between object reference and target scene tests a method's ability to adapt object appearance to local lighting.

Capture protocol. We capture all scenes indoors with a tripod-mounted camera so that the empty background and the composite share viewpoints. For each scene we first capture the empty background, then place the object at up to 8 distinct positions within the scene (median 3) and capture the composite at every position. Object reference images are captured separately under a different lighting environment than the target scene. Cameras include a Canon G7X, Galaxy smartphone, iPhone 15 Pro, and GoPro 10, with per-scene fixed white balance and exposure.

Scene diversity. The 49 scenes span living rooms, kitchens, bedrooms, offices, and corridors. Lighting ranges from single overhead fixtures to multi-source setups with colored smart bulbs, table lamps, and window light. We collect scenes with strong spatially varying illumination: visible cast shadows, color-temperature gradients, and localized highlights.

Object categories. The 62 objects span ten functional categories: *household goods* (step stool, soap, dispensers, containers, paper towels, cups; 11 objects), *electronics & appliances* (speakers, thermometers, air purifier, razor, VR headset, scanner, laptop, toaster; 11), *bags & wearables* (backpacks, caps, shoes, slippers, pouch, ear muffs, neck pillow; 11), *toys & textiles* (stuffed animals, baby toy, blankets, cushion; 6), *furniture* (table, stools, chair, ottoman; 5), *décor & lighting* (table lamp, diffuser, artificial plant, balloon, gift box; 5), *safety equipment* (fire extinguishers, traffic cone; 4), *kitchen & tableware* (cup, bowl, tumblers; 4), *stationery* (stapler, notebook, notepad; 3), and *food items* (bananas, orange; 2). Materials span plastic, ceramic, metal, wood, fabric, glass, paper, and fruit peel; shapes range from compact convex forms (e.g., orange, mug) to thin structures (e.g., lamp, traffic cone) and deformable items (e.g., blanket, stuffed animal). This diversity exercises illumination adaptation across a wide range of shapes and reflectance properties.

Annotation. Each triplet ships with an object segmentation mask from SAM2 [Ravi et al., 2025] (manually corrected) and a tight insertion bounding box.

Fig. 17 shows representative triplets across the lighting setups and object categories described above.

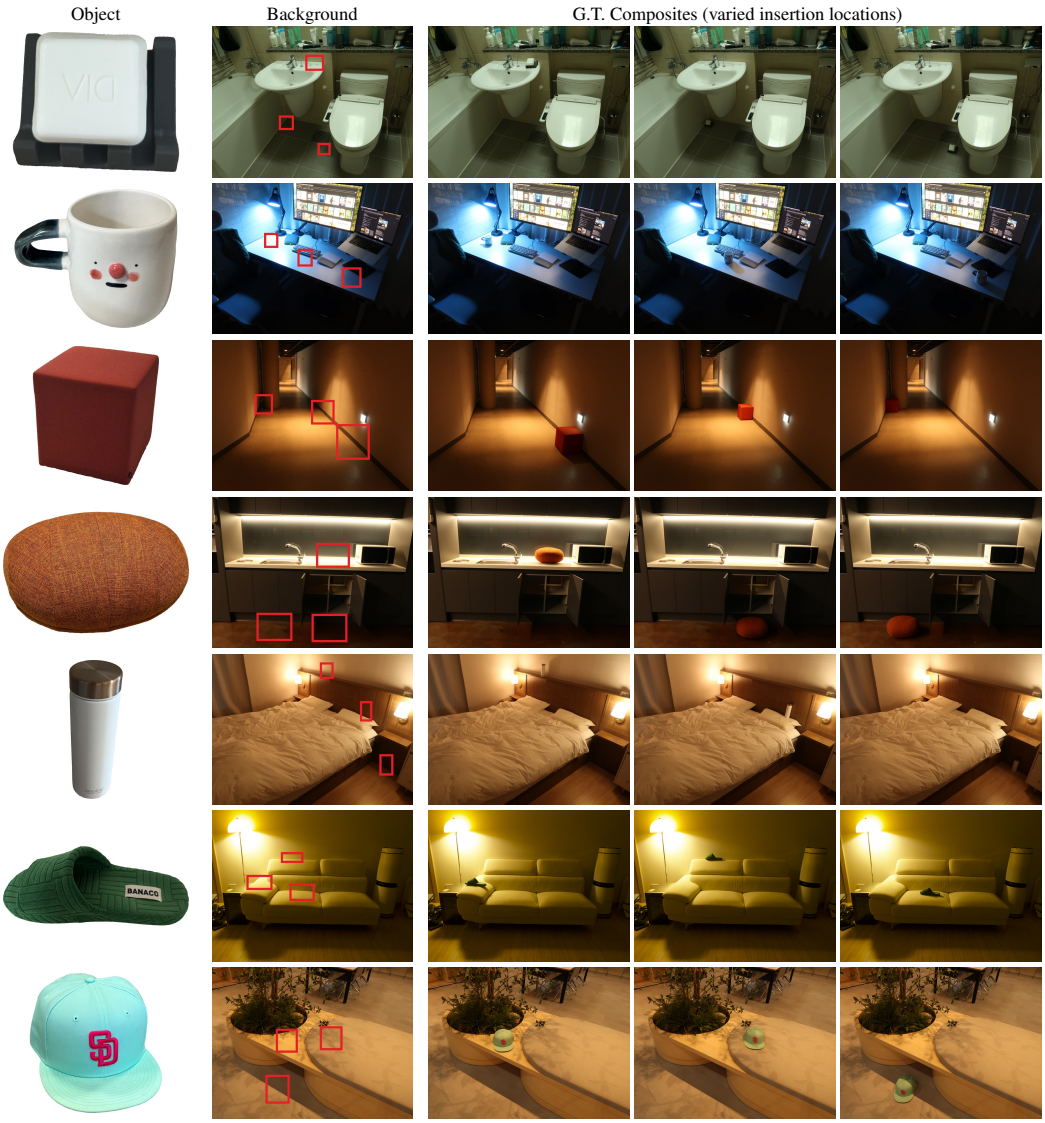


Figure 17: LOCO-bench samples. Each row shows one object and the matching empty background overlaid with three red bboxes marking the insertion locations, followed by the ground-truth composite at each bbox. Rows cover a tiled bathroom under overhead lighting, a cool-lit office desk, a warm-lit corridor, a kitchen counter under cove lighting, a bedroom with bedside lamps, a sofa under a floor lamp, and a planter bench in an indoor lobby.

A.7 Scene Preparation Details

This subsection details how the scene parameters consumed by the on-the-fly relighting pipeline (Sec. 3.1) are estimated and cached. Every step runs once per scene before training; at training time the augmentation only reads from this cache. Fig. 18 summarizes the pipeline.

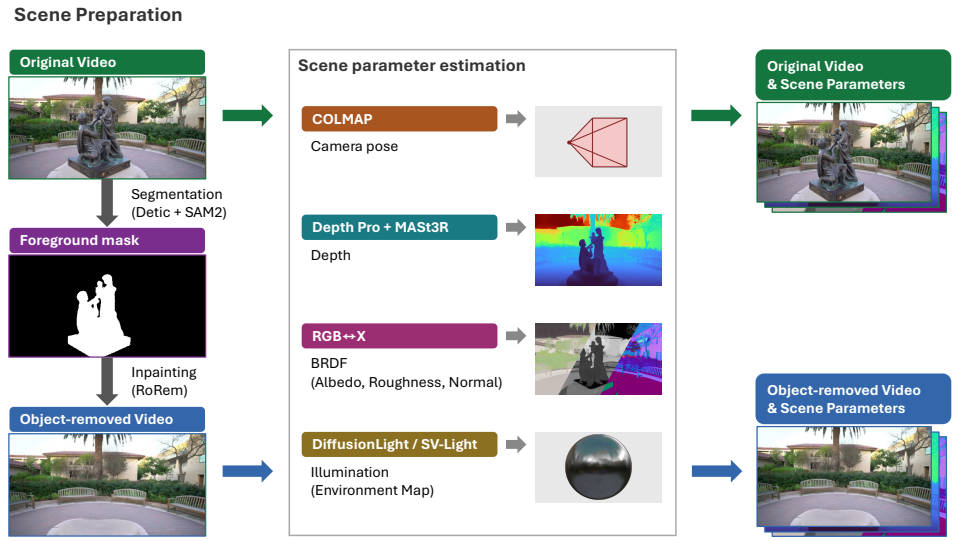


Figure 18: **Scene preparation pipeline.** Foreground objects are masked with Detic + SAM2 and inpainted out with RoRem to yield clean backgrounds. Camera poses are shared between the original and object-removed videos; depth and BRDF maps for the latter are completed in the removed regions, conditioned on the original estimates. All quantities are cached offline and reused throughout the on-the-fly relighting augmentation.

Foreground segmentation. We detect per-frame foreground objects with Detic [Zhou et al., 2022] using its open-vocabulary class set, then track each detected object across the whole video with SAM2 [Ravi et al., 2025] to obtain temporally consistent instance masks. Spurious detections and unreliable masks are removed by the filters detailed in Appendix A.9.

Background inpainting. Each tracked mask is dilated by a small margin so that the inpainter does not reconstruct the object from leaked boundary cues, then the masked region is inpainted by RoRem [Li et al., 2025], yielding an *object-removed* variant of the video. As a check for object re-hallucination, we measure the DINO similarity between the original object and its inpainted region. The similarity is 0.252, closer to unrelated background patches (0.203) than to the same object in another frame (0.691).

Geometry and reflectance estimation. Camera intrinsics and extrinsics are reconstructed once from the original video with COLMAP [Schönberger and Frahm, 2016] and shared with the object-removed variant. For depth, we first predict per-frame monocular depth with Depth Pro [Bochkovskii et al., 2025]. For each frame, we obtain correspondences to other frames with MASt3R [Leroy et al., 2024], estimate a candidate scale for each pair using RANSAC with fixed COLMAP poses, and take the median of these estimates as the frame's scale factor. For reflectance parameters, RGB↔X [Zeng et al., 2024] estimates albedo, roughness, and normal, which feed the simplified Disney BRDF detailed in Appendix A.1. For the object-removed frames, we complete the removed regions in the depth and BRDF maps with DepthLab [Liu et al., 2024b] and RGB↔X [Zeng et al., 2024], respectively, conditioned on the original-frame estimates. The completed maps are blended back using a dilated, blurred object mask, leaving the original estimates unchanged elsewhere.

Original scene illumination. Indoor and outdoor scenes require different illumination models, so we classify each scene with a vision-language model [Bai et al., 2025]. We sample up to 10 frames from the video, query the model on each for an indoor/outdoor label with a confidence score, and take the majority vote. We estimate a single HDR environment map with DiffusionLight [Phongthawee et al., 2024] for outdoor scenes, and per-pixel environment maps [Li et al., 2020] for indoor scenes. In both cases, we precompute the per-pixel shading induced by the estimated illumination (L_{orig}),

$$S_{\text{orig}}(x) = \int_{\Omega} L_{\text{orig}}(x, \omega_i)\, R(x, \omega_o, \omega_i)\, d\omega_i, \tag{10}$$

evaluated by discrete quadrature over the environment-map directions, using a single envmap outdoors and per-pixel envmaps indoors. Rather than caching the illumination maps directly, we cache S_{orig} and use it in the ratio-based illumination transfer of Eq. 3 (Sec. 3.1).

A.8 Training Data Statistics

We draw monocular videos from five sources: Mip-NeRF 360 [Barron et al., 2022], Tanks&Temples [Knapitsch et al., 2017], DL3DV [Ling et al., 2024], uCO3D [Liu et al., 2025], and our own casual captures. Our casual captures are handheld phone-camera videos recorded in everyday scenes, typically 30 seconds to 2 minutes long, without staging scenes specifically for spatially varying illumination. Fig. 6 of the main paper shows a raw triplet from these videos and its illumination-augmented counterpart. Each video is uniformly sampled to up to 100 frames and passed through the filtering pipeline of §A.9. Tab. 13 reports per-source counts after filtering. Representative augmented triplets and their cross-permutations appear in Figs. 19 and 20.

Table 13: Training data statistics per source. *Frames* counts unaugmented training triplets after filtering.

Source	Scenes	Objects	Frames	Avg. Views/Obj
MipNeRF360 [Barron et al., 2022]	9	39	1,980	50.8
Tanks&Temples [Knapitsch et al., 2017]	19	62	1,809	29.2
DL3DV [Ling et al., 2024]	5,731	21,373	536,132	25.1
uCO3D [Liu et al., 2025]	2,239	2,460	118,545	48.2
Casual Captures	51	247	12,433	50.3
Total	**8,049**	**24,181**	**670,899**	**27.7**

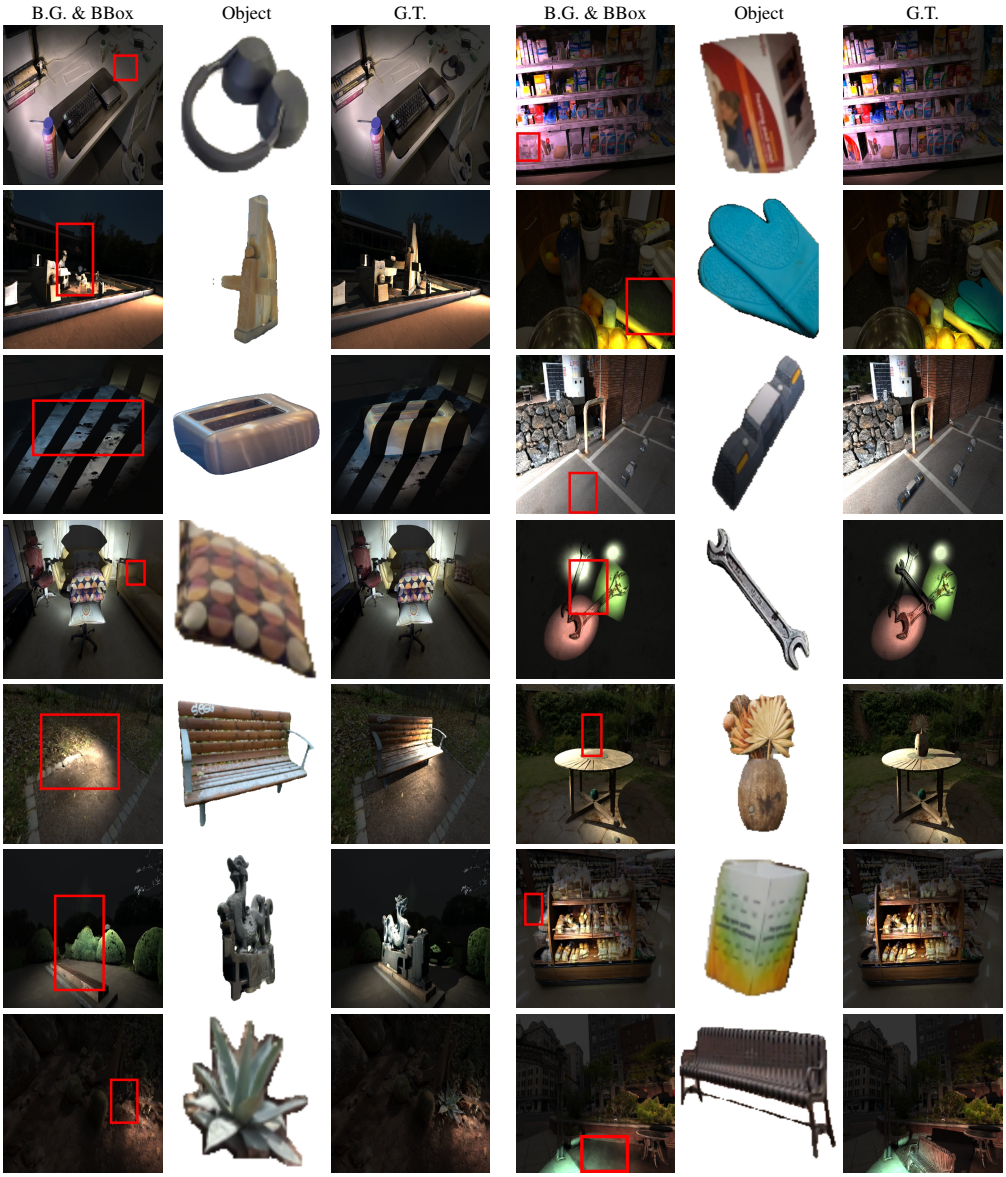


Figure 19: Examples of illumination-augmented training triplets $(I_{\text{bg}}, I_{\text{obj}}, I_{\text{comp}})$ produced by the pipeline in Sec. 3.1.

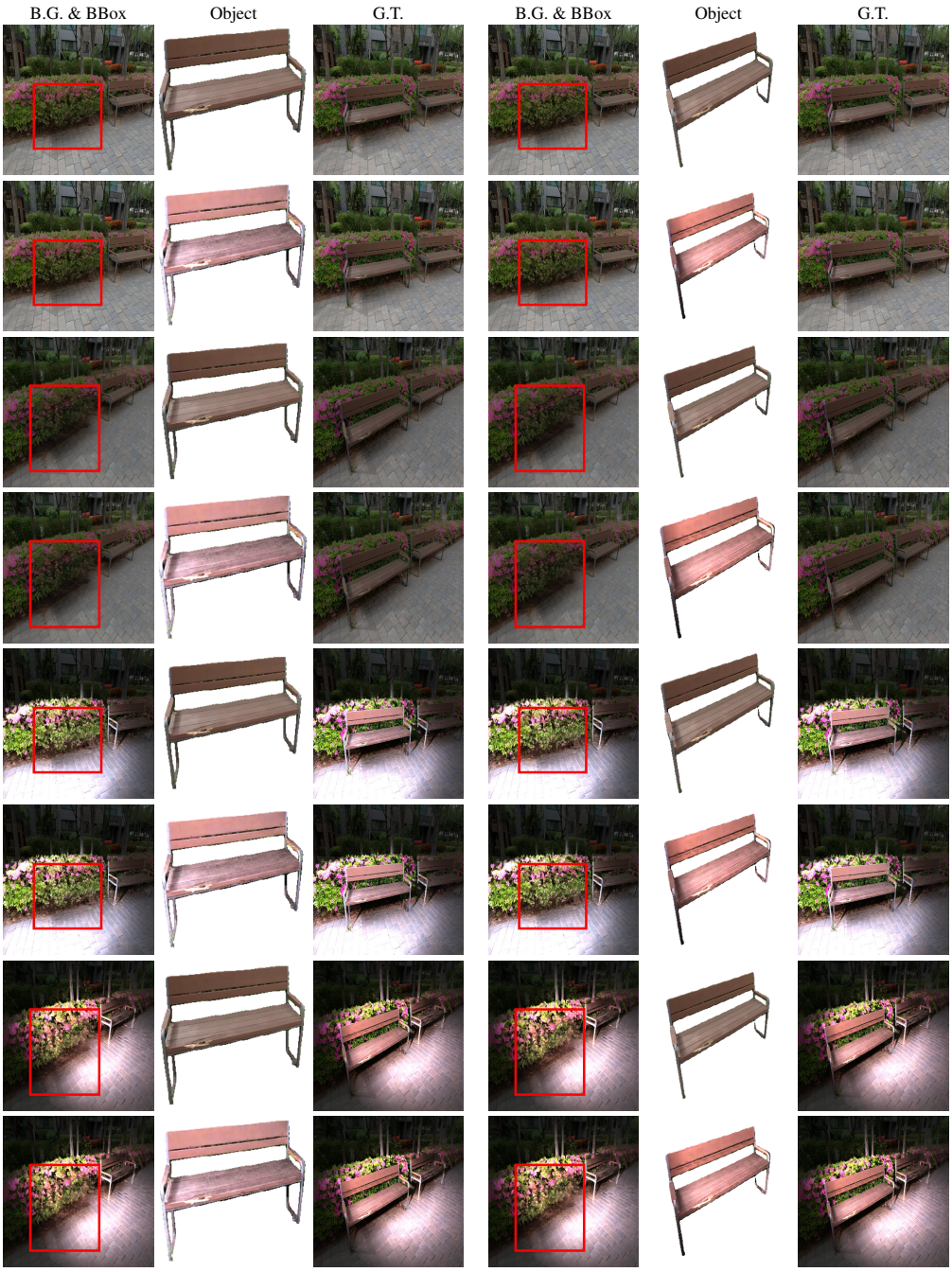


Figure 20: All 16 training triplets from the batch-permutation scheme with $V{=}2$ reference frames (Sec. 3.1). Each group of three columns shows one triplet: background with insertion bbox, input object, and target composite. Each of the $2V$ backgrounds is paired with an I_{obj} from any of the $2V$ composites, yielding $(2V)^2{=}16$ triplets.

A.9 Training Data Filtering

The automatically extracted training objects and scene estimates contain various artifacts, including inaccurate masks, tracking drift, degenerate geometry, and unstable normal predictions. We apply three stages of input filtering before training and additionally reject unreliable relighting outputs.

Stage 1: detection filtering. From Detic [Zhou et al., 2022] detections, we discard boxes whose area (normalized by image area) lies outside $[\tau_{\min}, \tau_{\max}]$, eliminating spurious tiny detections and scene-spanning false positives. Among the remaining detections, we keep the top N by confidence.

Stage 2: mask-shape filtering. We track each detected object across the video with SAM2 [Ravi et al., 2025] to obtain per-frame masks, then apply four complementary filters across all frames in which the object appears:

- *Small-object filter.* Discard frames where the mask covers less than τ_a of the image (object barely visible), and drop objects retained in fewer than N_f such frames.
- *Thin-object filter.* We remove elongated or sparse shapes (e.g., cables, table legs, picture-frame edges). For each frame we measure how densely the mask fills its bounding box, $e = A_{\text{mask}}/A_{\text{bbox}}$, and how stretched the box is, $r = \max(h, w)/\min(h, w)$, flagging the frame as thin if $e < \tau_e$ or $r > \tau_r$. We drop the object when more than half of its surviving frames are flagged.
- *Nested-object filter.* An object whose mask sits mostly inside another's is typically a sub-part of that object (e.g., a label on a bottle, a cushion on a sofa). For each ordered pair (i, j) we compute $|M_i \cap M_j|/|M_i|$ per frame and drop i if the ratio exceeds τ_{nest} in any frame.
- *Under-object filter.* When two objects share part of the image, the one positioned lower is often partially occluded by the upper one. For each ordered pair (i, j) whose bounding boxes overlap and where i is not nested in j, we drop i if its bounding-box center lies below j's, $y_c(B_i) > y_c(B_j)$ (image y grows downward), in any frame.

Stage 3: surface and feature filtering. Once scene decomposition and property estimation are complete, four further filters exploit estimated normals and learned features unavailable earlier:

- *Flat-surface filter.* We remove planar surfaces such as floors and walls, keeping only foreground objects. For each frame we compute the variance of estimated normals inside the object mask and flag the frame if the variance falls below τ_{flat}. We drop the object when more than half of its frames are flagged.
- *Noisy-normal filter.* We remove objects whose normal predictions are unstable. Using the same normal variance as above, we drop object frames whose value exceeds τ_{noisy}.
- *DINO consistency filter.* Mask tracking can drift to a different instance or background, especially when the object becomes partially occluded or small. To catch such failures, we take the frame with the largest mask (where the object is most clearly visible) as reference and compare every other frame's crop against it using DINOv2 [Siméoni et al., 2025] cosine similarity. We drop frames whose similarity falls below τ_{dino} and discard objects with fewer than 10 surviving frames.
- *Normal–ray consistency filter.* A surface visible to the camera should have its normal facing the camera; frames where many pixels violate this signal unreliable normal predictions. We compute the fraction of bounding-box pixels with $\mathbf{n}(x) \cdot (-\boldsymbol{\omega}_o(x)) < 0$ (normal pointing away from the camera) and drop frames whose fraction exceeds τ_{nr}.

Finally, we cap each scene at 8 objects, keeping those with the largest total mask area summed across surviving frames.

Relighting quality filtering. We discard degenerate relighting outputs, including holes or speckles in illuminated regions, light leaks, flat-shading artifacts, and cases where the foreground object is brightened but the surrounding background remains barely illuminated, leaving insufficient contextual lighting cues.

On DL3DV, detection filtering removes 91.1% of raw boxes, mask-shape filtering removes 53.1% of tracked objects, and surface filtering removes 8.0% of the remaining objects. The DINO, normal–ray, and relighting quality filters remove 38.2%, 4.0%, and 33.7% of frames, respectively.

A.10 Limitations

LOCO adapts the inserted object's shading to local lighting at the insertion location, but several limitations remain. We illustrate failure cases in Fig. 21.

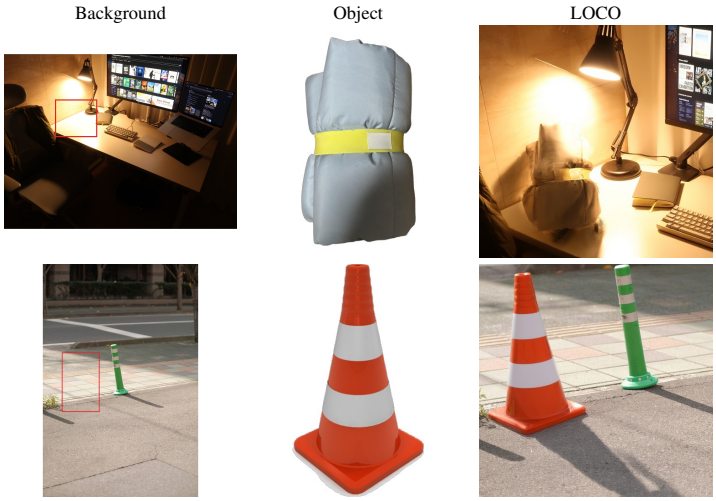


Figure 21: Failure cases of LOCO. *Top:* object affordance is ignored. A soft cushion-like object is placed upright on a desk, an unstable pose; the natural placement is to lay it flat so that it does not topple. *Bottom:* Outdoor scenes follow a different lighting regime. A distant sun illuminates the scene from a single direction regardless of the location, resulting in parallel cast shadows. Our virtual-spotlight augmentation models point-source illumination and does not reproduce this parallel-shadow structure.

Resolution. Our 512×512 generation resolution limits fine-scale object appearance and lighting detail. A possible extension is to first generate the illumination-adapted crop at 512×512 and then refine the inserted region at higher resolution, conditioned on the original object and background. We have not evaluated this extension and leave it to future work.

Articulated and deformable objects. LOCO treats the inserted object as rigid: its appearance adapts to the location, but joints do not bend and shape does not deform. For example, a cup adapts to a tilted surface, whereas a standing figure does not re-pose to sit on a chair. Affordance-driven re-posing is a separate problem from illumination adaptation.

Complex geometry and materials. The augmentation relies on monocular depth and approximate BRDF estimates, which are unreliable for thin structures, highly specular surfaces, and transparent materials, leading to relighting artifacts. Glossy objects are partially supported through the microfacet specular term of the BRDF, but small high-frequency reflections are unreliable. Thin or complex objects are more susceptible to segmentation and depth errors; the filters of §A.9 protect training quality but leave such objects under-represented in the training data.

Outdoor scenes. We focus on indoor scenes, where multiple localized emitters such as lamps and windows produce spatially varying illumination over short distances. Outdoor scenes exhibit a different lighting structure: a distant sun acts as the dominant source, so light rays arrive at nearly the same angle throughout the scene and cast shadows are parallel [Hold-Geoffroy et al., 2017, Zhu et al., 2021]. Our virtual spotlights model point-source illumination and may therefore not fully capture this parallel-shadow structure. Covering both indoor and outdoor lighting within a unified framework remains an open problem [Dastjerdi et al., 2023]; extending our augmentation to outdoor scenes is left to future work.

A.11 Broader Impact

LOCO extends object compositing to scenes with spatially varying lighting, with applications in content creation, interior-design visualization, and augmented reality. The same capability can be misused for deceptive imagery and fraud; LOCO does not introduce a new manipulation capability, but improves the local-lighting plausibility of an existing task already served by diffusion-based compositing. As generative quality rises, the community should continue to invest in detection and provenance tools.

A.12 Image Attribution

The outdoor failure case in Fig. 21 (bottom row) uses third-party assets:

- **Background:** "A green and white tube on a sidewalk" by Mak (`https://unsplash.com/photos/a-green-and-white-tube-on-a-sidewalk-Js1NAy3LQUQ`), Unsplash License (`https://unsplash.com/license`).
- **Object (traffic cone):** "Cones PNG 13323" from `https://pngimg.com/image/13323`, pngimg License (`https://pngimg.com/license`).

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 clearly state our three contributions: (1) studying object compositing under spatially varying illumination, (2) a physically grounded illumination-augmentation pipeline that turns monocular videos into training data, and (3) the LOCO-bench benchmark of physically captured relit composites with ground-truth relit objects. These are supported by experiments in Section 4.

   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: We discuss limitations in Section 5 and Appendix A.10.

   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 (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - 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. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - 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.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

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: This paper does not include theoretical results or proofs. Our contributions are primarily empirical.

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, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, 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 (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide architecture details, training hyperparameters, and dataset construction procedures in the main paper (Section 3 and Section 4) and supplementary material (Appendix A.1).

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: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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: An anonymized code and LOCO-bench is provided as a supplementary zip; LOCO-bench images are downscaled to a long edge of 1280 pixels to fit within the supplementary size limit.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Training details including optimizer, learning rate, and batch size are provided in Section 4 and Appendix A.1.

   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 that is necessary to appreciate the results and make sense of them.
   - The 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: We do not report error bars because training and evaluating diffusion models across multiple random seeds is computationally costly. We instead report results across multiple datasets, metrics, and user studies on large benchmark sets (2,280 Laval SV HDR and 760 LOCO-bench).

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- 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. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We report the GPU type and training time in Appendix A.1.

   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 amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into 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: We have reviewed the NeurIPS Code of Ethics and our research conforms with it in every respect.

   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 (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

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

    Answer: [Yes]

Justification: We discuss broader impacts in Appendix A.11, including potential misuse for generating deceptive imagery and mitigation strategies.

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 their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

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 (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: Our model is a task-specific compositing model rather than a general-purpose image generator. It does not pose significant risks beyond those already present in existing diffusion-based methods.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite the original sources of all datasets and pretrained models used in this work. Third-party images shown in qualitative figures are credited with their licenses in Appendix A.12.

    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 (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's 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: We introduce LOCO-bench and document its construction, statistics, and intended use in Section 4 and Appendix A.6.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

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 (if any)?

    Answer: [Yes]

    Justification: We conduct a pairwise user study comparing LOCO against each baseline; the protocol, judgment criteria, and presentation format are described in Appendix A.5.

    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, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - 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 Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The user study collected only pairwise preference judgments over rendered images from volunteer participants, with no personal data collected and no compensation.

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. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

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? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are not used as a component of the core method in this research.

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 our LLM policy in the NeurIPS handbook for what should or should not be described.