Supplement: Photorealistic Object Insertion with Diffusion-Guided Inverse Rendering

In the supplement, we provide additional ablation for diffusion guidance (Sec. A), implementation details (Sec. B), additional results on user study and tone-mapping (Sec. C), and discuss broader impact (Sec. D). Please refer to the *accompanied video* for more qualitative results.

A Diffusion Personalization and Score Distillation

An intuitive approach to understanding the diffusion guidance is to directly visualize the text-to-image generation result of the diffusion model. In this section, we provide additional analysis and ablative visualization on our design choices of LoRA personalization (Fig. S1) and concept preservation (Fig. S2).

Personalizing diffusion model. Due to the high stochasticity in the diffusion denoising process, the images generated by a pre-trained diffusion model often cannot be tailored to a specific input image. However, in the setting of using the diffusion model for solving the inverse rendering problem of the given scene, it is important to preserve the key context (e.g., shapes, lighting, and shadowing effects) from the unseen input background image. In Fig. S1, we visualize the effect of LoRA personalization. After personalization, the diffusion model can generate images in a similar domain to the target scene.

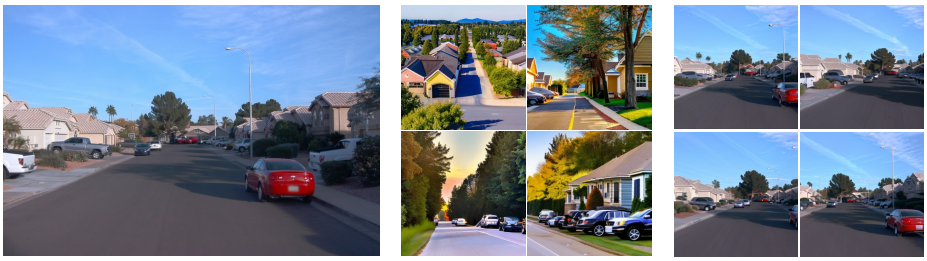

Fig. S1: Illustrations of text-to-image generation results. "SD21 w/o LoRA" shows our best-effort prompting results for outdoor street scenes from off-the-shelf Stable Diffusion 2.1. "SD21 w/ LoRA" directly uses the training prompt *"a scene in the style of sks rendering"*. LoRA personalization enables generating images in a similar domain to the target scene.

Fig. S2: Text-to-image generation results with prompt *"**a black SUV car** in the style of sks rendering"*. Concept preservation can facilitate the high-quality generation of both the object and the background scene.

Concept preservation. For the task of virtual object insertion, it is important to ensure the personalized diffusion model does not completely overfit the input background image, and can generalize when inserting new objects into the scene through additional text conditions. Personalizing diffusion model (DM) along with some generated class images is inspired by the original DreamBooth paper [11] which shows that adding in-class images for concept preservation can avoid concept drift and improve the output diversity.

In Fig. S2, we visualize the text-to-image generation results with and without using concept preservation. The results show that only personalizing with the input image does not generalize well when adding additional concepts into the text prompt – Diffusion model does not faithfully follow the additional prompt to synthesize images with "a black SUV car" in it. The reddish car color is heavily affected by the car shown in the input background image. The results of concept preservation show the benefits of retaining the appearance of newly inserted objects.

LDS loss design. The original SDS loss tends to generate over-saturated images. Recent work [7, 16] observes that the classifier score $\boldsymbol{\delta} = \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t, \boldsymbol{c}) - \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t, \varnothing)$ dominates the optimization direction, and directly distilling the classifier score can provide much better quality. Our LDS loss (Eq. **??**) is inspired by the classifier score distillation and adapts it to the personalized diffusion model. The delta term in LDS loss is calculated between LoRA fine-tuned conditional denoising term $\boldsymbol{\epsilon}_{(\boldsymbol{\theta}+\boldsymbol{\Delta W})}(\boldsymbol{z}_t, t, \boldsymbol{c})$ and non-adapted unconditional denoising term $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t, \varnothing)$. The intuition of not "using the LoRA fine-tuned model in both terms" is to encourage gradient towards the personalized model with scene-specific knowledge, while not overly biased by the small amount of training data used in personalization. We empirically observe our LDS loss is more stable and leads to better quality. We ablate this design choice in Fig. S3 and defer rigorous theoretical understanding of this loss to further work.

Fig. S3: Ablation on unconditional denoising term in LDS loss.

B Implementation Details

Diffusion Model We use Stable Diffusion 2.1 as our pre-trained diffusion models throughout experiments. To reduce memory overhead and accelerate the training, we use PyTorch's FP16 mixed floating point training for diffusion models by default. The whole optimization can be run on a GPU with more than 12GB VRAM.

Rendering and image formation. The differentiable rendering framework is built on Mitsuba 3 [6]. We use 128 samples per pixel, and spawn 4 rays each for multiple importance sampling (MIS) [1] of BSDF and emitters. The output resolution is 256×384, which we crop and bilinearly upsample to 512×512 to feed into the personalized diffusion model.

The tone-mapping function for the input image is often unknown, and thus we use the default Reinhard tone-mapping [10] for the inserted virtual object $\mathbf{I}_{\text{fg}} = \text{Reinhard}(\mathbf{I}_{\text{HDR}})$. As described in the main paper, the rendered pixels are then passed into the single-channel optimizable tone correction function $\tilde{\mathbf{I}}_{\text{fg}} = f(\mathbf{I}_{\text{fg}}; \boldsymbol{\theta}_{\text{fg}})$. The tone correction function $f(\cdot)$ is an optimizable spline curve that differentiably maps real values from the range $[0, 1]$ to $[0, 1]$, which aims to learn the residual of the default Reinhard tone-mapping. The shadow ratio is directly multiplied onto the tone-mapped input image, and thus we do not apply additional tone-mapping and directly pass it into the tone correction function $\tilde{\boldsymbol{\beta}}_{\text{shadow}} = f(\boldsymbol{\beta}_{\text{shadow}}; \boldsymbol{\theta}_{\text{shadow}})$. All notations in the main paper and supplement operate in linear RGB space following graphics conventions, and we finally convert with gamma correction ($\gamma = 2.2$) to produce sRGB output.

Environment map fusion. Following the description in the main paper, we initialize two sets of optimizable Spherical Gaussian (SG) parameters and compute two separate environment maps, $\mathbf{L}_{\text{fg}}, \mathbf{L}_{\text{shadow}} \in \mathbb{R}^{H \times W \times 3}$, to light the foreground inserted object and cast shadows respectively.

The additional capacity can improve quality and stabilize the training in the early stage of optimization, and we aim to progressively fuse them into a single environment map at the end of optimization. Let $\tilde{\mathbf{L}}_{*} \in \mathbb{R}^{H \times W}$ denote the luminance of each environment map, we compute the fused environment map

$\mathbf{L}_{\text{fused}} \in \mathbb{R}^{H \times W \times 3}$ by adjusting the luminance of the foreground environment map:

$$\mathbf{L}_{\text{fused}} = \mathbf{L}_{\text{fg}} \cdot \frac{\tilde{\mathbf{L}}_{\text{fused}}}{\tilde{\mathbf{L}}_{\text{fg}}} \tag{1}$$

where the target luminance of the fused environment map is computed by blending the two environment maps:

$$\tilde{\mathbf{L}}_{\text{fused}} = (1 - \mathbf{r}) \cdot \tilde{\mathbf{L}}_{\text{fg}} + \mathbf{r} \cdot \tilde{\mathbf{L}}_{\text{shadow}} \tag{2}$$

$$\mathbf{r} = \frac{\tilde{\mathbf{L}}_{\text{fg}}}{\max(\tilde{\mathbf{L}}_{\text{fg}})} \cdot \frac{\tilde{\mathbf{L}}_{\text{shadow}}}{\tilde{\mathbf{L}}_{\text{fg}} + \tilde{\mathbf{L}}_{\text{shadow}}}. \tag{3}$$

Here $\mathbf{r} \in \mathbb{R}^{H \times W}$ is the per-pixel blending ratio which encourages the fused luminance to favor $\mathbf{L}_{\text{shadow}}$ at high luminance pixels and $\mathbf{L}_{\text{fg}}$ at low luminance pixels.

As the optimization progresses, the fused environment map $\mathbf{L}_{\text{fused}}$ is linearly scheduled to replace the two environment maps $\mathbf{L}_{\text{fg}}, \mathbf{L}_{\text{shadow}}$ for the rendering of $\mathbf{I}_{\text{fg}}$ and $\boldsymbol{\beta}_{\text{shadow}}$:

$$\mathbf{I}_{\text{fg}} = \text{PathTrace}(\mathcal{X}, \mathbf{L}_{\text{fg}}^{'}, D) \tag{4}$$

$$\boldsymbol{\beta}_{\text{shadow}} = \frac{\text{PathTrace}(\mathcal{X} \cup \mathcal{P}, \mathbf{L}_{\text{shadow}}^{'}, 1)}{\text{PathTrace}(\mathcal{P}, \mathbf{L}_{\text{shadow}}^{'}, 1)} \tag{5}$$

where the scalar value s is scheduled to linearly increase from 0 to 1:

$$\mathbf{L}_{\text{fg}}^{'} = s \cdot \mathbf{L}_{\text{fused}} + (1 - s) \cdot \mathbf{L}_{\text{fg}} \tag{6}$$

$$\mathbf{L}_{\text{shadow}}^{'} = s \cdot \mathbf{L}_{\text{fused}} + (1 - s) \cdot \mathbf{L}_{\text{shadow}} \tag{7}$$

such that $\mathbf{L}_{\text{fg}}^{'} = \mathbf{L}_{\text{shadow}}^{'} = \mathbf{L}_{\text{fused}}$ at the end of optimization.

3D assets. We use 6 licensed 3D car models from Turbosquid and 3DModels.org for experiments on Waymo outdoor street scenes, and 11 assets from Sketchfab and PolyHaven for PolyHaven HDRI scenes.

Running time overhead. The total running time is about 26 min (∼13 min for LoRA DM finetuning + ∼13 min for distillation sampling) on an RTX A6000 GPU with FP16 mixed precision inference.

C Additional Results

In this section, we provide further experimental details and additional results.

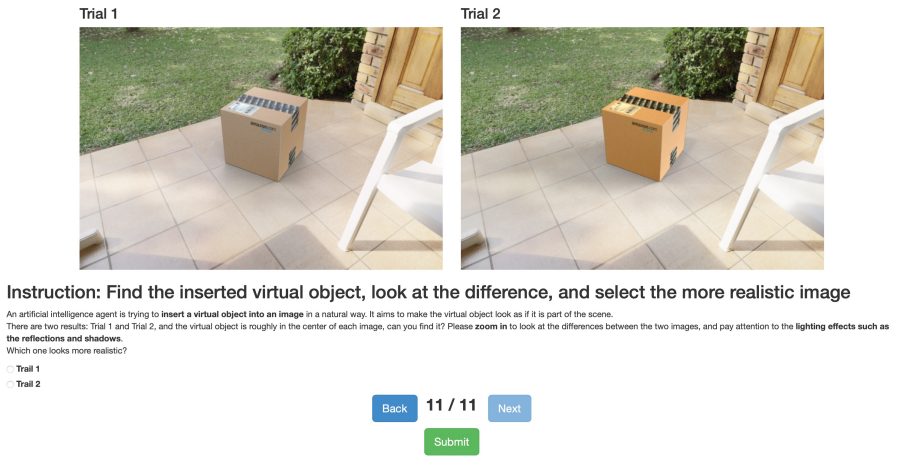

Fig. S4: Visualization of interface for user study.

C.1 User Study

User study is a standard approach for assessing perceptual realism of virtual object insertion [2–4, 13, 15]. Following prior works, we conduct a user study on Amazon Mechanical Turk to compare with prior methods and ablate our design choices.

User interface. Participants receive a pair of two object insertion results: one generated using our proposed method, the other one using a baseline approach. Participants are instructed to evaluate the differences between the two images, focus on the lighting effects of the inserted objects, and select the image they deemed to be *more realistic*:

> An artificial intelligence agent is trying to insert a virtual object into an image in a natural way. It aims to make the virtual object look as if it is part of the scene. There are two results: Trial 1 and Trial 2, and the virtual object is roughly in the center of each image, can you find it? Please zoom in to look at the differences between the two images, and pay attention to the lighting effects such as the reflections and shadows. Which one looks more realistic?

The participants are required to use a 24-inch or larger monitor to view the results, and the images are shuffled in a random order to prevent bias. The user interface is visualized in Fig. S4.

Statistics. We invited 9 different users for each experiment setting, and repeated each experiment 3 times. For benchmarking experiments, there are 48 scenes on the Waymo dataset to compare with 4 baselines, and 11 scenes on the

Table S1: User study: benchmark on Waymo outdoor street scenes. We report the percentage of *images* and *user selections* that our method is preferred over baselines. A preferred percentage $> 50\%$ indicates Ours outperforming baselines.

% images Ours is preferred	Daytime		Twilight	Night	All scenes
	Sunny	Cloudy			
DiffusionLight [8]	$80.4 \pm 12.2\%$	$68.9 \pm 20.4\%$	$55.6 \pm 11.1\%$	$71.4 \pm 14.3\%$	$70.8 \pm 3.6\%$
Hold-Geoffroy *et al.* [5]	$60.8 \pm 6.8\%$	$66.7 \pm 13.3\%$	$74.1 \pm 25.7\%$	$85.7 \pm 24.7\%$	$68.8 \pm 2.1\%$
NLFE [13]	$80.4 \pm 6.8\%$	$73.3 \pm 11.5\%$	$44.4 \pm 11.1\%$	$52.4 \pm 21.8\%$	$67.4 \pm 3.2\%$
StyleLight [12]	$76.5 \pm 15.6\%$	$91.1 \pm 10.2\%$	$66.7 \pm 22.2\%$	$66.7 \pm 8.2\%$	$77.8 \pm 12.6\%$

% user selection Ours is preferred	Daytime		Twilight	Night	All scenes
	Sunny	Cloudy			
DiffusionLight [8]	$63.4 \pm 4.0\%$	$61.2 \pm 7.6\%$	$53.9 \pm 8.7\%$	$59.3 \pm 8.1\%$	$60.3 \pm 1.9\%$
Hold-Geoffroy *et al.* [5]	$56.4 \pm 1.6\%$	$54.1 \pm 7.5\%$	$61.7 \pm 9.8\%$	$68.8 \pm 12.1\%$	$58.5 \pm 1.9\%$
NLFE [13]	$65.4 \pm 1.1\%$	$58.3 \pm 4.1\%$	$50.6 \pm 1.2\%$	$56.6 \pm 2.4\%$	$59.1 \pm 1.2\%$
StyleLight [12]	$60.6 \pm 8.2\%$	$68.9 \pm 8.5\%$	$61.7 \pm 6.5\%$	$59.8 \pm 10.3\%$	$63.3 \pm 6.0\%$

Table S2: User study: benchmark on PolyHaven scenes. We report the percentage of *images* and *user selections* that our method is preferred over baselines. A preferred percentage $> 50\%$ indicates Ours outperforming baselines.

Methods	% images Ours is preferred	% user selection Ours is preferred
DiffusionLight [8]	$66.7 \pm 5.2\%$	$57.2 \pm 0.6\%$
Wang *et al.* [14]	$84.8 \pm 18.9\%$	$66.3 \pm 1.5\%$
StyleLight [12]	$75.8 \pm 5.2\%$	$60.6 \pm 5.2\%$

PolyHaven dataset to compare with 3 baselines. This results in a number of $48 \times 4 \times 9 \times 3 = 5184$ and $11 \times 3 \times 9 \times 3 = 891$ user selections for each dataset. For the ablation study, we randomly select a subset of 18 scenes from the Waymo dataset to reduce cost, and compare with 6 ablated versions of our method. The number of user selections for the ablation study is $18 \times 6 \times 9 \times 3 = 2916$. The total number of user selections for all experiments is 8991.

Metrics and additional results. Our primary evaluation metric is the percentage of *images* that our method was preferred over the baseline, following [13]. Specifically, for each sample, we collect the binary selection from 9 different users and do majority voting from the 9 users to determine which method is more preferred on this *sample*. The majority voting can efficiently filter the effects of random users, and we report this as the primary metric in the main paper. The full experiments are repeated three times to calculate the mean and standard deviation. We additionally report the standard deviation in Table S1, S2, S3. Note that the standard deviation reflects the consistency in user evaluations *after* majority vote, where a high standard deviation suggests the compared methods performed on par on some of the examples.

We also report the percentage of *user selections* that our method is preferred over the baselines in Table S1, S2, S3. Our method consistently outperforms baseline methods and ablated versions of our method.

Table S3: User study: ablation study on Waymo outdoor street scenes. We report the percentage of *images* and *user selections* that our method is preferred over baselines. A preferred percentage $> 50\%$ indicates Ours outperforming the ablated versions.

Methods	% images Ours is preferred	% user selection Ours is preferred
Ours (dataset update)	$85.2 \pm 25.7\%$	$67.9 \pm 10.2\%$
Ours (SDS [9])	$74.1 \pm 12.8\%$	$64.0 \pm 5.0\%$
Ours (SDS [9] w/o LoRA)	$90.7 \pm 8.5\%$	$71.6 \pm 9.6\%$
Ours (w/o concept preservation)	$64.8 \pm 14.0\%$	$56.0 \pm 5.8\%$
Ours (w/o tone curve)	$68.5 \pm 12.8\%$	$56.2 \pm 6.5\%$
Ours (w/o env. map fusion)	$66.7 \pm 9.6\%$	$57.6 \pm 6.6\%$

(a) w/o optimizing tone curves **(b)** w/ optimizing tone curves

Fig. S5: Qualitative ablation on tone-mapping curve optimization. The optimizable tone-mapping curve provides the capacity and flexibility to match the scale and color of the shadows. (The visualized foreground curve considers gamma correction $\gamma = 2.2$.)

C.2 Additional Qualitative Results

Fig. S7 and Fig. S8 show the additional qualitative comparison against other baseline methods. Our method consistently performs well in various background images with challenging light conditions, while the baseline models often fail to capture the correct lighting direction or intensity scale. We also include more insertion examples in Fig. S9 and Fig. S10. Video examples can be found on the project page.

C.3 Tone-mapping Curve

In Fig. S5, we visualize the optimized tone-mapping curve and ablate the effect of the optimizable tone-mapping curves in our method. Comparing the results, the optimizable tone-mapping curve can effectively adjust the color and scale of

the rendered shadow, and be blended more naturally in the background image. The foreground curve learns a residual from Reinhard tonemapping and is often close to identity mapping.

C.4 Failure Case Analysis

In Fig. S6, we show examples for the limitations mentioned in Sec. **??**. *a) Shiny reflection.* Our insertion of a shiny sphere correctly captures the general highlight direction, but cannot get all high-frequency details due to the limitations of SG lighting; *b) Color drift.* The inserted dustbin is relatively brighter than the reference. This is because DM's data prior tends to assume a "recycle dustbin" with a bright green color; *c) Double shadowing.* Our current method does not handle double shadowing with local occlusion, which can be improved when combined with 3D reconstruction methods.

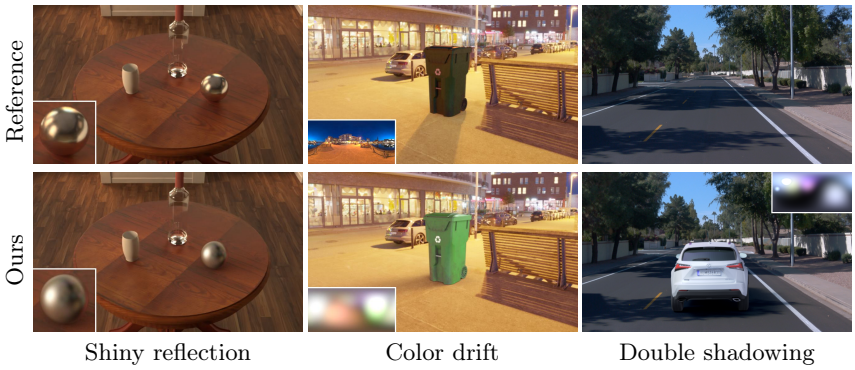

Fig. S6: Failure case examples.

D Discussion

Broader impact. This paper introduces a novel approach to producing virtual object insertion in images leveraging the power of diffusion models and inverse rendering techniques. It could benefit digital content creation by providing filmmakers and game developers with a powerful tool to create novel scenarios and reducing costly manual editing. Its application in AR and VR can enhance user experiences, making digital interactions feel more natural and engaging. On a broader scale, this work contributes to the field of computer vision and graphics, and showcases the potential of combining powerful diffusion models with classic rendering techniques.

Similar to other photorealistic image editing technologies such as "deep fakes", there is also the potential for misuse, *e.g.* it might potentially be used to create misleading content and propagate misinformation. Related technology on identifying and filtering out such content can mitigate these negative applications.

References

1. Black, M.J., Anandan, P.: The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. Computer vision and image understanding **63**(1), 75–104 (1996) 3
2. Gardner, M.A., Hold-Geoffroy, Y., Sunkavalli, K., Gagné, C., Lalonde, J.F.: Deep parametric indoor lighting estimation. In: ICCV. pp. 7175–7183 (2019) 5
3. Gardner, M.A., Sunkavalli, K., Yumer, E., Shen, X., Gambaretto, E., Gagné, C., Lalonde, J.F.: Learning to predict indoor illumination from a single image. arXiv preprint arXiv:1704.00090 (2017) 5
4. Garon, M., Sunkavalli, K., Hadap, S., Carr, N., Lalonde, J.F.: Fast spatially-varying indoor lighting estimation. In: CVPR. pp. 6908–6917 (2019) 5
5. Hold-Geoffroy, Y., Athawale, A., Lalonde, J.F.: Deep sky modeling for single image outdoor lighting estimation. In: CVPR. pp. 6927–6935 (2019) 6, 10
6. Jakob, W., Speierer, S., Roussel, N., Nimier-David, M., Vicini, D., Zeltner, T., Nicolet, B., Crespo, M., Leroy, V., Zhang, Z.: Mitsuba 3 renderer (2022), https://mitsuba-renderer.org 3
7. Ling, H., Kim, S.W., Torralba, A., Fidler, S., Kreis, K.: Align your gaussians: Text-to-4d with dynamic 3d gaussians and composed diffusion models (2023) 2
8. Phongthawee, P., Chinchuthakun, W., Sinsunthithet, N., Raj, A., Jampani, V., Khungurn, P., Suwajanakorn, S.: Diffusionlight: Light probes for free by painting a chrome ball. In: ArXiv (2023) 6, 10, 11
9. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. arXiv (2022) 7
10. Reinhard, E., Stark, M., Shirley, P., Ferwerda, J.: Photographic tone reproduction for digital images. ACM Trans. Graph. **21**(3), 267–276 (jul 2002). https://doi.org/10.1145/566654.566575, https://doi.org/10.1145/566654.566575 3
11. Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., Aberman, K.: Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation (2022) 2
12. Wang, G., Yang, Y., Loy, C.C., Liu, Z.: Stylelight: Hdr panorama generation for lighting estimation and editing. In: European Conference on Computer Vision (ECCV) (2022) 6, 10, 11
13. Wang, Z., Chen, W., Acuna, D., Kautz, J., Fidler, S.: Neural light field estimation for street scenes with differentiable virtual object insertion. In: ECCV (2022) 5, 6, 10
14. Wang, Z., Philion, J., Fidler, S., Kautz, J.: Learning indoor inverse rendering with 3d spatially-varying lighting. In: ICCV (2021) 6, 11
15. Wang, Z., Shen, T., Gao, J., Huang, S., Munkberg, J., Hasselgren, J., Gojcic, Z., Chen, W., Fidler, S.: Neural fields meet explicit geometric representations for inverse rendering of urban scenes. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2023) 5
16. Yu, X., Guo, Y.C., Li, Y., Liang, D., Zhang, S.H., Qi, X.: Text-to-3d with classifier score distillation (2023) 2

Fig. S7: Additional visual comparisons on inserting virtual car assets into Waymo driving scenes.

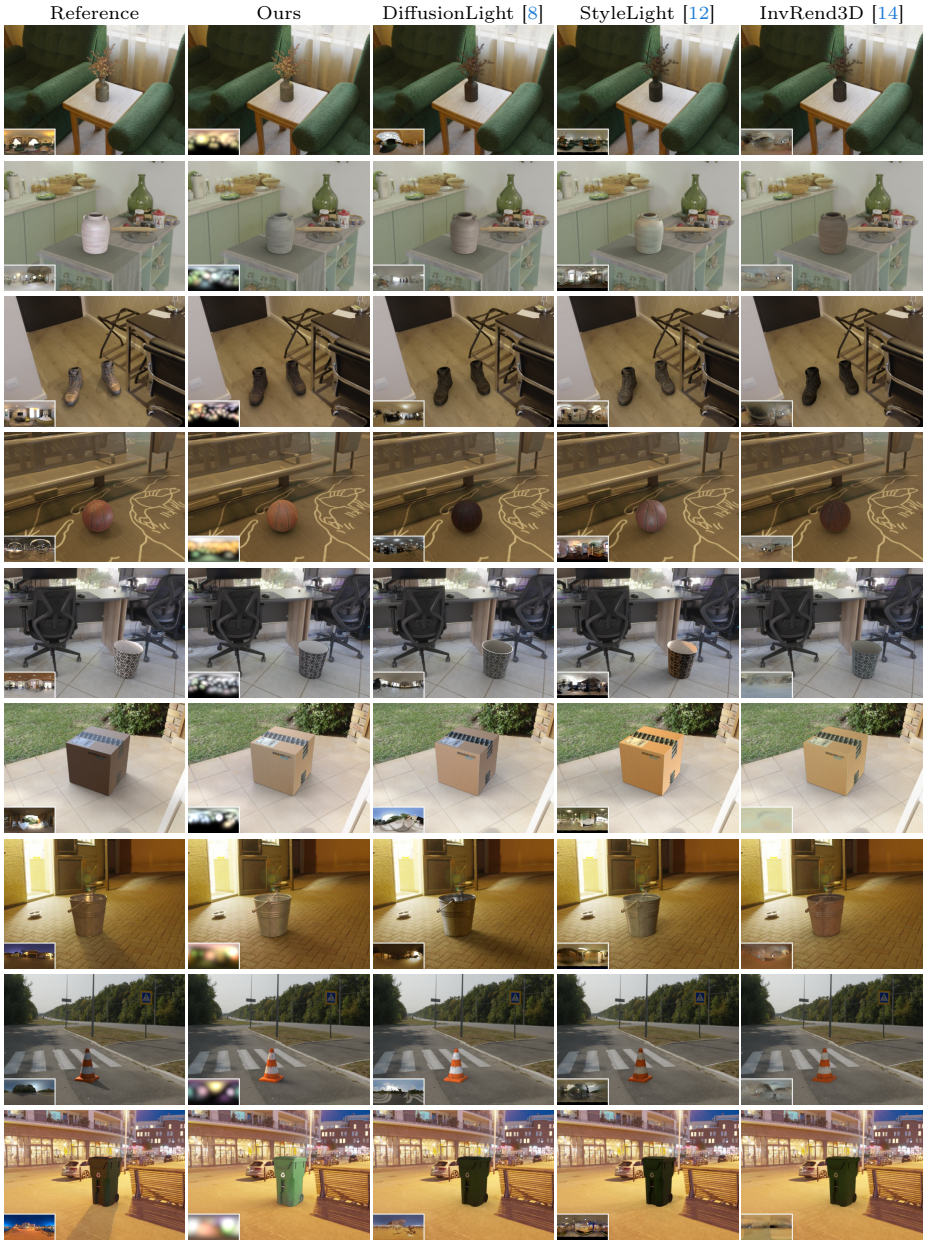

Fig. S8: Additional visual comparisons on inserting objects into cropped HDRIs from PolyHaven.

Fig. S9: Additional car insertion examples on Waymo driving scenes.

Fig. S10: Additional object insertion examples on cropped PolyHaven HDRIs.