NeAR: Coupled Neural Asset–Renderer Stack

Supplementary Material

A. More Implementation Details

A.1. Different Single-Image Relighting Paradigms

To further elucidate the architectural advantages of NeAR, we present a structured comparison of three representative paradigms in Fig. 1.

2D-based Intrinsic Decomposition [23, 54]): As shown in Fig. 1(a-b), these methods decompose a single image into 2D PBR maps (e.g., albedo, roughness, normals). However, lacking an underlying 3D representation, they struggle to disentangle view-dependent specular highlights from base color, and fail to support novel-view rendering and accurate shadow modeling.

Single Image to 3D Textured Mesh [5, 6, 63]): Typical 3D generative models (Fig. 1(c)) follow a decoupled paradigm, where asset construction (geometry and PBR textures) is fully separated from rendering. While compatible with standard graphics pipelines (e.g., Blender), they rely on highly ill-posed PBR inversion, often leading to material ambiguity (e.g., misclassifying metallic surfaces as diffuse).

Our Coupled NeAR Stack: In contrast, NeAR adopts a "homogenize-then-synthesize" paradigm. By lifting single-image inputs under arbitrary lighting into a Lighting-Homogenized SLAT (LH-SLAT), we construct an illumination-invariant neural asset that serves as a stable rendering substrate, preserving cues of geometry, uniform lighting, and material interactions. On top of this, a coupled neural renderer learns to interpret these homogenized latents, enabling the synthesis of complex light–material interactions.

A.2. Implementation Details

Training Details. We conduct all training experiments on four NVIDIA H100 80GB HBM3 GPUs.

In the **LH-SLAT Reconstruction & Generation phase** (§3.3), we fine-tune a rectified flow model equipped with LoRA [15] to normalize shaded SLATs from arbitrary images into LH-SLATs. The goal of this phase is to learn a mapping, f_θ, that transforms light-dependent shaded SLAT representations into light-homogenized counterparts. To achieve this efficiently while preserving the prior knowledge of the original flow model f_s [45], we initialize LoRA using PEFT [29]. We configure LoRA with a rank of 512 and a scaling factor, further integrating rslora [17] to enhance training stability. LoRA adaptors are applied to the query, key, value, and output projection modules within the attention mechanism. We optimize the model using AdamW [27] with a learning rate of 1×10^{-4}. This training phase takes approximately two days to complete.

In the **Relightable Neural 3DGS Synthesis phase** (§3.5), we employ the AdamW optimizer with a batch size of 48. The learning rate is warmed up linearly to 1×10^{-4} over the first 5K steps, followed by a cosine decay schedule. We perform end-to-end joint training on the IAD, LAD, and the Lighting-Aware tokenizer (denoted as E_l). To accelerate training, we leverage Flash-Attention 3 [37] and gsplat [49]. The model is trained for 500K iterations across all loss components, requiring approximately 10 days. Additionally, we investigated the incorporation of geometric constraint losses, specifically normal and depth losses. However, we observed that adding these regularizations degraded both convergence speed and rendering quality, suggesting a trade-off between geometric constraints and rendering fidelity within the 3DGS framework [4, 50].

Inference Details. Given a single input image I_{in} with unknown lighting and a target high dynamic range (HDR) environment map E, our inference pipeline proceeds as follows. Since our method decouples geometry generation from relighting, we first reconstruct a 3D mesh m from I_{in} using Hunyuan3D 2.1 (HY3D 2.1) [63] with default settings. This mesh is then voxelized to provide coordinates for the structurally sparse voxel feature SLAT.

Following Trellis [45], we utilize the pre-trained SLAT flow model f_s to generate an initial shaded SLAT Z_s from I_{in}. Note that Z_s inherently contains arbitrary lighting information from the input image. To remove these lighting effects, we concatenate Z_s with noise (matching Z_s in shape) along the channel dimension and feed the result into our fine-tuned corrective model f_θ to yield the Lighting-Homogenized SLAT (LH-SLAT).

Subsequently, for the target lighting, we pre-process the environment map E (as detailed in Sec. 3.5.2) and encode it into a lighting condition embedding C_L using the Lighting-Aware tokenizer $\mathcal{E}_l$. The IAD module then processes the LH-SLAT to extract intrinsic features h. Simultaneously, the LAD $\mathcal{D}_E$ integrates the viewing direction encoding e^v and the lighting condition C_L to predict the 3D Gaussian attributes for the specific view and lighting (Eq. 5). Finally, we render the relit HDR image I^{hdr}_{target} via gsplat [49]. To align the visual output with standard rendering engines like Blender, we apply AgX tone mapping[3] to convert the HDR result into a low dynamic range (LDR) image.

[3] https://github.com/iamNCJ/simple-ocio

A.3. Network Architectures

Register Tokens. Apart from the lighting-aware tokenizer $\mathcal{E}_l$, and consistent with [45], our method primarily employs Transformer networks. As depicted in Fig. 4, the IAD $\mathcal{D}_I$ comprises 3D shifted window multi-head self-attention (3D-SW-MSA) and a feed-forward network (FFN). Addressing the limitation of the naive 3D-SW-MSA design in [45], which computes attention solely within local windows and neglects inter-window information exchange, we introduce learnable register tokens. These tokens interact with all windows via 3D multi-head cross-attention (3D-MCA), serving as a global information bridge to facilitate the model's learning of global context. The lighting-aware decoder $\mathcal{D}_E$ receives intrinsic features h, view encoding e^v, register tokens, and lighting encoding to generate lighting-dependent features h^v. Register tokens and lighting encoding are injected into the network via 3D-MCA. Here, h and e^v are added in a voxel-wise manner to determine which lighting encoding tokens should be attended to under the current viewpoint. The ablation study on the interaction order of viewpoint and lighting information is illustrated in Fig. 7 and Tab 4.

Loss Functions. For the relightable neural 3DGS synthesis stage, we optimize the model using a composite objective function $\mathcal{L}_{\text{total}}$. This objective is a weighted sum of three primary reconstruction components—HDR reconstruction ($\mathcal{L}_{\text{recon}}$), physically-based material supervision ($\mathcal{L}_{\text{pbr}}$), and shadow-casting ($\mathcal{L}_{\text{shadow}}$)—along with regularization terms for Gaussian primitives:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{recon}} + \lambda_{\text{pbr}}\mathcal{L}_{\text{pbr}} + \lambda_{\text{shadow}}\mathcal{L}_{\text{shadow}} + \lambda_{\text{vol}}\mathcal{L}_{\text{vol}} + \lambda_{\alpha}\mathcal{L}_{\alpha}. \tag{7}$$

In our experiments, we set the weighting hyperparameters to $\lambda_{\text{pbr}} = 0.3$, $\lambda_{\text{shadow}} = 0.5$, $\lambda_{\text{vol}} = 10,000$, and $\lambda_{\alpha} = 0.001$.

Reconstruction Loss ($\mathcal{L}_{\text{recon}}$). We formulate $\mathcal{L}_{\text{recon}}$ to ensure high-fidelity HDR rendering. Before calculating perceptual metrics, we apply AgX tone mapping to both the rendered HDR image $I^{\text{hdr}}_{\text{target}}$ and the ground-truth $I^{\text{hdr}}_{\text{gt}}$, yielding their LDR counterparts $\hat{I}_{\text{target}}$ and $\hat{I}_{\text{gt}}$. The loss combines an L1 distance in the logarithmic domain for HDR consistency, along with SSIM and LPIPS losses on the tonemapped LDR images for perceptual quality:

$$\begin{aligned}\mathcal{L}_{\text{recon}} = \; & \mathcal{L}_1(\log(I^{\text{hdr}}_{\text{target}} + 1), \log(I^{\text{hdr}}_{\text{gt}} + 1)) \\ & + 0.2(1 - \text{SSIM}(\hat{I}_{\text{target}}, \hat{I}_{\text{gt}})) \\ & + 0.2\text{LPIPS}(\hat{I}_{\text{target}}, \hat{I}_{\text{gt}}).\end{aligned} \tag{8}$$

PBR and Shadow Supervision. To guide the model towards physically plausible decomposition, we impose direct constraints on the intermediate PBR feature maps. The material loss $\mathcal{L}_{\text{pbr}}$ supervises the base color (I^b), roughness (I^r), metallic (I^m), and shading (I^s) maps against their ground truths:

$$\mathcal{L}_{\text{pbr}} = \mathcal{L}_1(I^b, I^b_{gt}) + \mathcal{L}_1(I^r, I^r_{gt}) + \mathcal{L}_1(I^m, I^m_{gt}) + \mathcal{L}_1(I^s, I^s_{gt}). \tag{9}$$

Similarly, $\mathcal{L}_{\text{shadow}}$ employs an $\mathcal{L}_1$ loss to ensure the geometric consistency of cast shadows under novel lighting conditions.

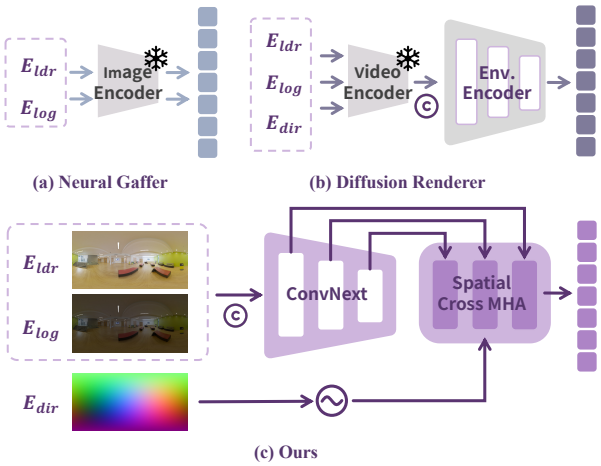

Figure 8. Compared to existing HDRI encoding methods, we bind directional information with multi-scale features using positional encoding.

Regularization. To prevent the degeneration of Gaussian primitives (e.g., becoming too large or too opaque) during optimization [45], we incorporate a volumetric loss $\mathcal{L}_{\text{vol}}$ and an opacity loss $\mathcal{L}_{\alpha}$:

$$\begin{aligned}\mathcal{L}_{\text{vol}} &= \frac{1}{LK}\sum_{i=1}^{L}\sum_{k=1}^{K}\prod \boldsymbol{s}_i^k + \frac{1}{LK}\sum_{i=1}^{L}\sum_{k=1}^{K}\prod \hat{\boldsymbol{s}}_i^k, \\ \mathcal{L}_{\alpha} &= \frac{1}{LK}\sum_{i=1}^{L}\sum_{k=1}^{K}(1-\alpha_i^k)^2.\end{aligned} \tag{10}$$

These terms are calculated across the L active voxels, with each voxel predicting K Gaussian primitives. Specifically, $\mathcal{L}_{\text{vol}}$ regularizes the scale components $\boldsymbol{s}$ from the IAD and $\hat{\boldsymbol{s}}$ from the LAD simultaneously.

Lighting Tokenizer. As illustrated in Fig. 8, the lighting tokenizer $\mathcal{E}_l$ is primarily designed to process and inject lighting information into the network for relighting purposes, while also effectively perceiving rotations in the ambient lighting. Similar to Neural Graffer and Diffusion Renderer, as depicted in Fig. 8 (a), our approach leverages the E_{hdr} and E_{log} components of the environment map E to provide lighting color characteristics. Neural Graffer encodes the environment map into an image latent space via a pre-trained image VAE, whereas Diffusion Renderer employs a video VAE model to compress E_{ldr}, E_{log}, and E_{dir}

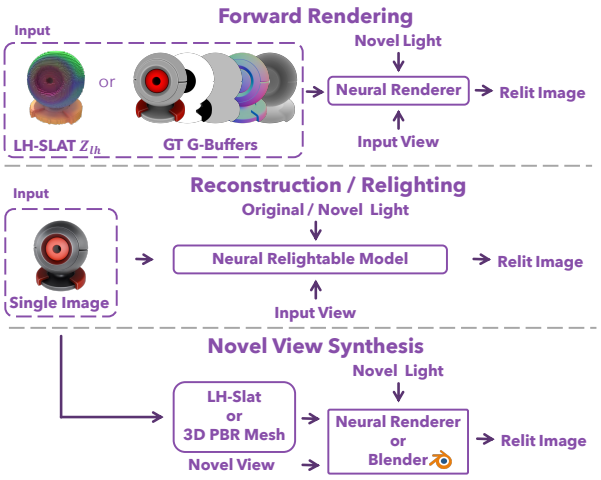

Figure 9. Schematic illustration of four distinct sub-tasks.

into a video latent space of consecutive frames, thereby accommodating subsequent image or video diffusion model training.

The lighting-aware tokenizer, $\mathcal{E}_l$, is primarily designed to process and inject lighting information into the network to enable relighting while also effectively perceiving rotations in the environment map. Similar to Neural Graffer and Diffusion Renderer, as depicted in Fig. 8, our approach leverages the E_{hdr} and E_{log} components of environmental illumination to provide lighting color features. Neural Graffer encodes the environment map into the image latent space using a pretrained image VAE's encoder. Diffusion Renderer, in contrast, employs a video VAE model to compress E_{ldr}, E_{log}, and E_{dir} into a video latent space of consecutive frames, thus accommodating subsequent image or video diffusion model training. As shown in Fig. 8(c), the design of $\mathcal{E}_l$ aims to facilitate the injection of lighting information from the LH-SLAT into relightable 3D Gaussian Splatting (GS). This design addresses two key challenges: First, different materials require sensitivity to varying resolutions of lighting information. For instance, highly rough surfaces require only low-resolution environment maps, whereas high-metallicity surfaces with low roughness necessitate high-resolution maps. To address this, we utilize ConvNext [25] to extract multi-resolution features from the lighting pyramid and employ a spatial attention mechanism to compute attention scores and exchange information between these resolutions. Second, the model should accurately perceive rotations of the environment map. Neural Graffer requires deforming the environment map itself. Our approach, similar to Diffusion Renderer, can rotate the illumination by adjusting the environment light direction map, E_{dir}, requiring only the application of a rotation matrix to the direction vector. However, Diffusion Renderer relies on an additionally trained environment encoder (Env. Encoder). Our method employs a direction-encoding-aware spatial cross-multihead attention (Spatial Cross MHA) to guide visual features at different resolutions using directional information. It combines multi-scale feature fusion to preserve both detailed and global information and utilizes a RoPE+RMSNorm transformer layer for efficient sequence modeling, refer to Fig. 4. This allows the complex HDRI lighting information to be encoded into conditional tokens suitable for cross-attention, providing high-quality lighting conditions for the renderer. Abstractly, we model the environment map as a set of light source tokens, each encoded with absolute direction vector positional information and multi-scale features. Subsequently, the Lighting-Aware Decoder (LAD), $\mathcal{E}_l$, can efficiently determine the relevance of each token to the current viewpoint by leveraging viewpoint direction encoding.

A.4. Experiments Setup

Fig. 9 shows the quantitative evaluation setup for four subtasks—forward rendering, reconstruction, relighting, and novel view synthesis—as summarized in Tab. 1. It illustrates how methods, given different input modalities (e.g., LH-SLAT, G-buffers, single images, or 3D assets), are processed under varying viewpoints and lighting via neural rendering or relightable models to produce relit images.

B. More Results

B.1. Additional Comparisons

Qualitative Evaluation. We provide comprehensive visual comparisons to further substantiate the effectiveness of our method. Figures 14 and 15 illustrate additional results for single-image reconstruction under diverse illumination conditions. For single-image relighting with unknown input lighting, we present extended comparisons in Figures 16 and 17. Notably, our method recovers significantly more accurate shadows and specular highlights compared to existing 2D diffusion-based relighting models [16, 23, 52, 54], which often struggle with physical consistency.

Comparison with 3D Generation Baselines. We also conduct detailed comparisons against state-of-the-art 3D generation methods capable of producing PBR materials [3, 5, 6, 63]. As shown in Fig. 18, our approach demonstrates superior material disentanglement, yielding highlights and tonal values that align closely with the ground truth. To ensure a fair comparison and isolate material quality from geometric failures, we provide the baselines with a fixed frontal view and evaluate the rendered output from the same perspective. This setup mitigates the potential for geometric collapse or severe artifacts in baseline methods, focusing the evaluation on rendering and relighting fidelity.

B.2. Additional Visualization Results

Texture Style Transfer. As shown in Fig. 10, given the target-style image in the second column and an arbitrary

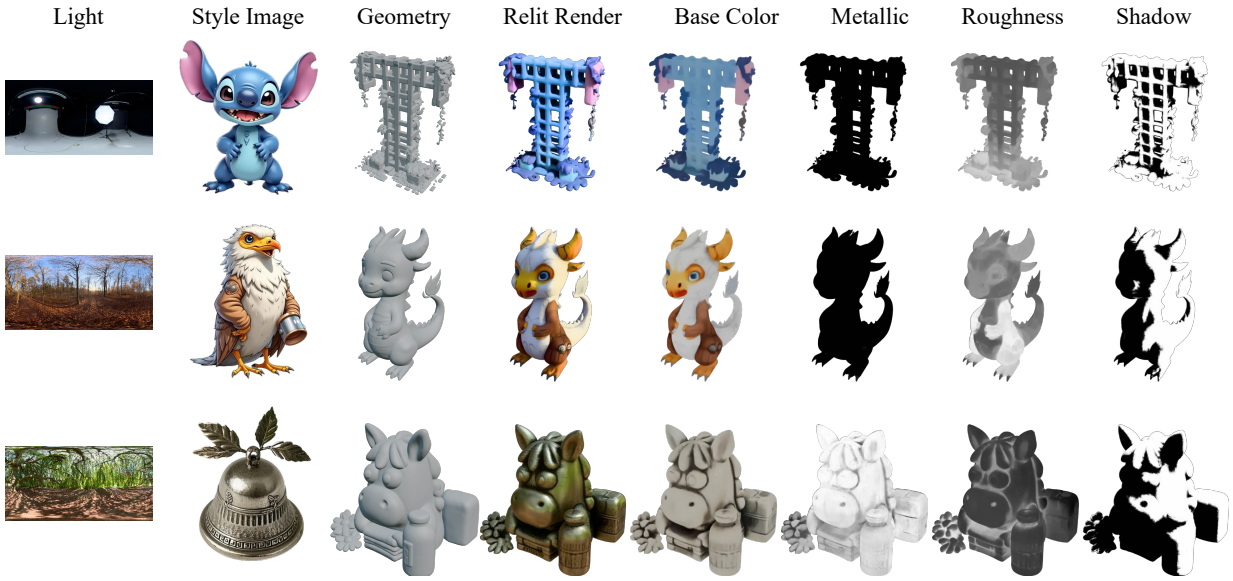

Figure 10. **Texture Style Transfer.** Given the geometry and a target-style image as guidance, our method can generate semantically consistent stylized textures and support photorealistic neural relighting.

geometry, we convert the geometry as coordinates, while deriving the LH-SLAT from the stylized image using the flow models f_s and f_θ. This enables semantically consistent style transfer and material estimation. For instance, the dragon's mouth in the second row corresponds to the beak style in the reference image, and the third-row horse metalness map matches the metallic appearance implied by the target style. Moreover, benefiting from LH-SLAT, we can produce photorealistic neural relighting (in the third column) under a given illumination condition, for example, the green reflections from the tree and metallic highlights in the third row.

Real-World Data. As shown in Fig. 11, to demonstrate generalization to real-world data, NeAR successfully removes baked-in lighting from both internet images (rows 1–2) and real mobile photos (row 3), enabling effective relighting under novel views. It further exhibits strong robustness in handling specular highlights and capturing lighting directionality (rows 2–3).

PBR Material and Shadow Decomposition. Leveraging a single input image and a target environment map, our pipeline enables high-fidelity, relightable 3D Gaussian Splatting synthesis with support for multi-view rendering. In Fig. 19, we visualize the decomposed PBR material maps (Albedo, Roughness, Metallic) and the generated shadow maps. These visualizations explicitly demonstrate the effectiveness of our physically-based supervision signals (discussed in Sec. 3.5.3) in achieving clean and plausible material decomposition.

Figure 11. Real-world relighting results from single images.

C. Discussion

C.1. Limitations and Future Work

Despite the robust performance of our framework in generalized single-image relightable 3D Gaussian synthesis, several challenges remain that define coordinates for future research.

Fine-grained Detail Preservation. As illustrated in Fig. 13 (Left), the reconstruction of high-frequency textures—such as small text—is currently hindered by the feature compression pipeline. The semantic features from DINOv2 [31] undergo substantial downsampling to match the resolution of the LH-SLAT voxel grid. This bottleneck inevitably leads to the loss of intricate details. Future iterations will explore multi-scale feature refinement

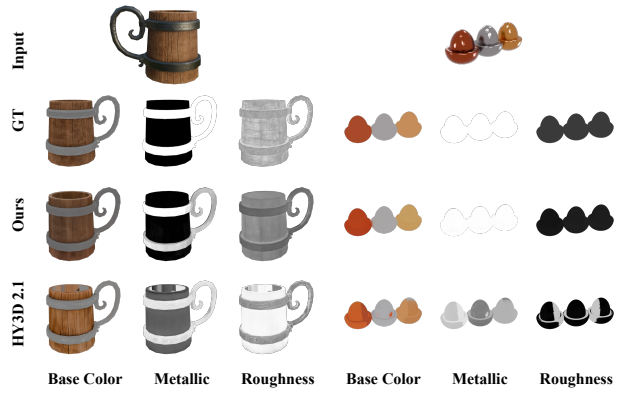

Figure 12. Visual comparison of PBR material estimation with HY3D2.1 and our method.

Figure 13. **Failure Cases. Left:** High-frequency details (e.g., text) are blurred due to voxel resolution constraints and VAE compression loss. **Right:** Transparent objects exhibit checkerboard artifacts. While 3DGS theoretically supports alpha blending, data scarcity in current datasets leads to inaccurate density estimation for refractive surfaces.

or sparse high-resolution voxel structures to better preserve these fine-grained elements.

Complex Material Modeling. Handling transparent or highly refractive materials remains a significant challenge. While the alpha-blending mechanism of 3DGS natively supports semi-transparency, our model's ability to represent such surfaces is heavily dependent on the training data distribution. Due to the scarcity of high-quality transparent objects in current large-scale datasets, the model occasionally fails to densify Gaussians sufficiently, resulting in the artifacts shown in Fig. 13 (Right). Incorporating specialized physics-based transmission losses or curated transparent object datasets could mitigate this issue.

C.2. Scalability and Generalization

A core strength of our proposed framework is its inherent scalability across data and architecture, which ensures its long-term viability as a foundation for 3D generative tasks.

Data Scalability. Our modular, multi-stage design allows for independent scaling of different components. Stage 1 (Lighting Homogenization) directly benefits from increasing data volume and diversity, as it learns to suppress complex baked-in illumination—a task that scales effectively with broader data distributions. In contrast, Stage 2 (Lighting-aware Synthesis) is highly efficient due to its feed-forward nature, primarily requiring diversity in material properties and lighting conditions rather than sheer volume.

Architectural Scalability. All core components of our pipeline are based on Transformer architectures, which offer a predictable path for capacity scaling. As demonstrated in our ablation studies (see Tab. 2), increasing model capacity consistently improves rendering quality, particularly for complex specular effects. Furthermore, the LH-SLAT representation is spatially scalable; increasing the voxel resolution allows the model to capture more complex geometries and lighting interactions. This flexibility enables a practical trade-off between inference speed and rendering fidelity depending on the target application.

Overall, the capacity of our model to generalize across diverse object categories and lighting conditions validates our core design philosophy. Our framework validates the effectiveness of jointly designing neural rendering and neural asset stacks, providing a robust and extensible path toward high-fidelity, relightable 3DGS synthesis from arbitrary single images.

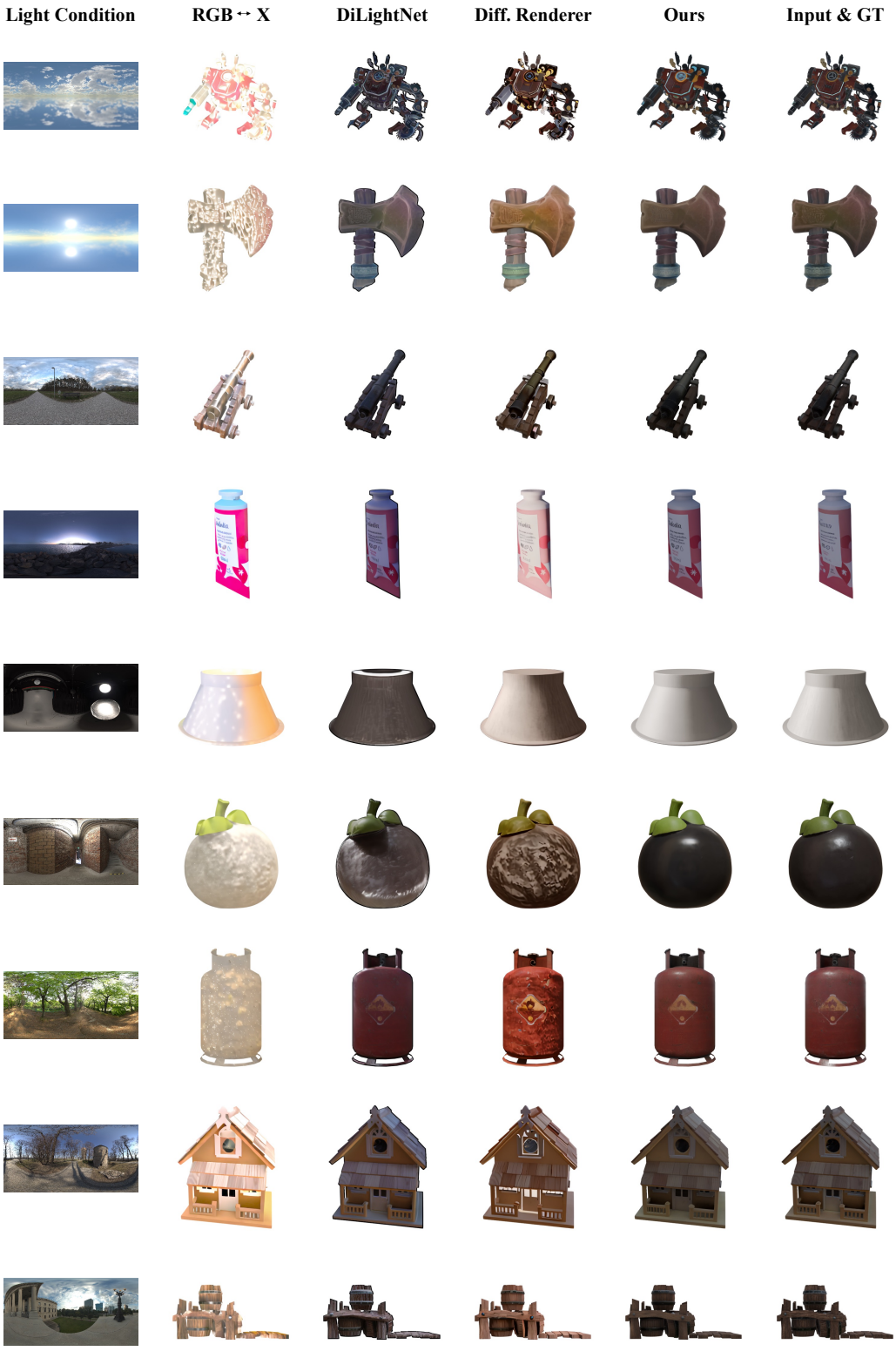

Figure 14. Additional qualitative results for single-image reconstruction under random illumination.

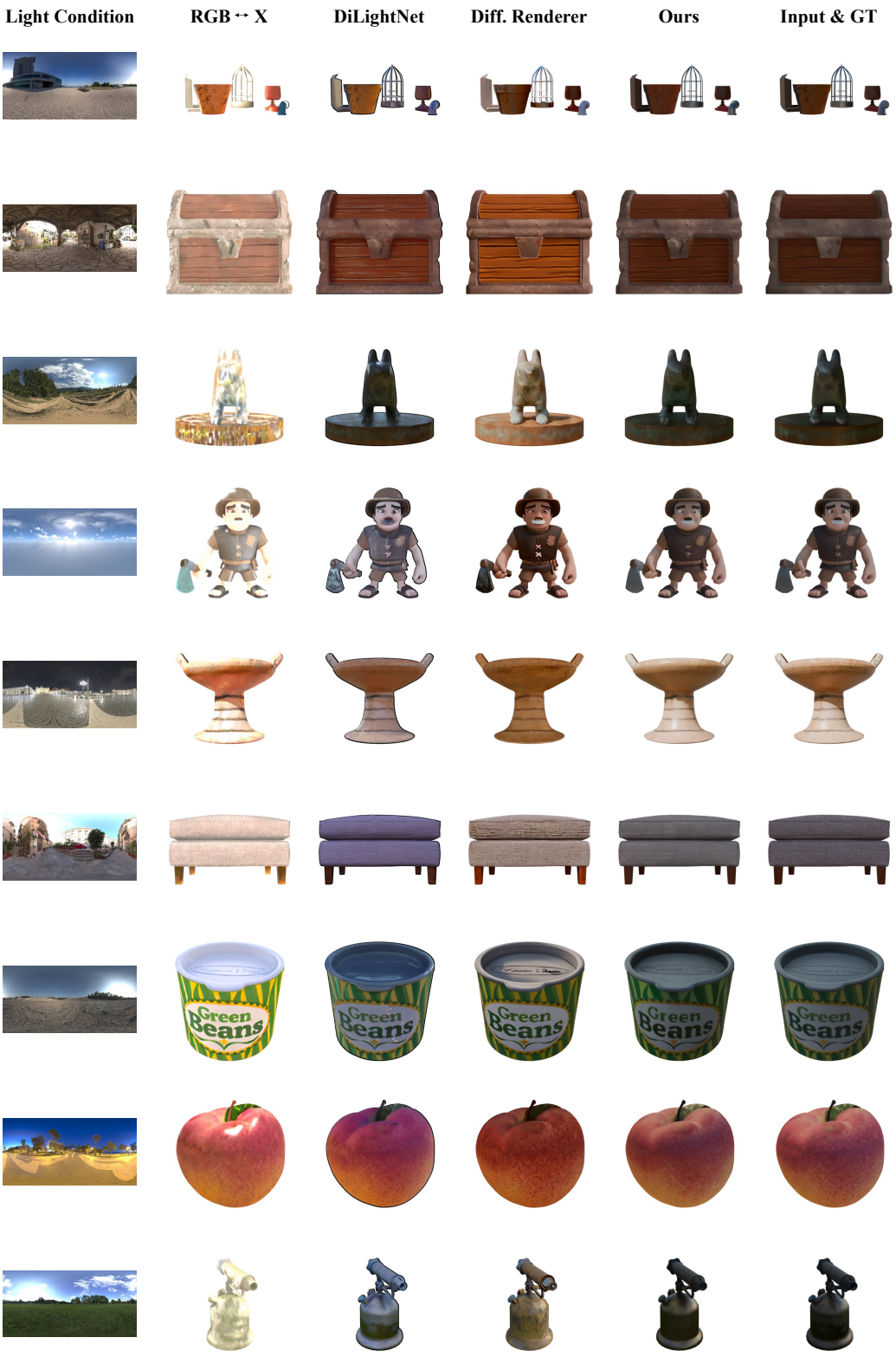

Figure 15. Additional qualitative results for single-image reconstruction under random illumination.

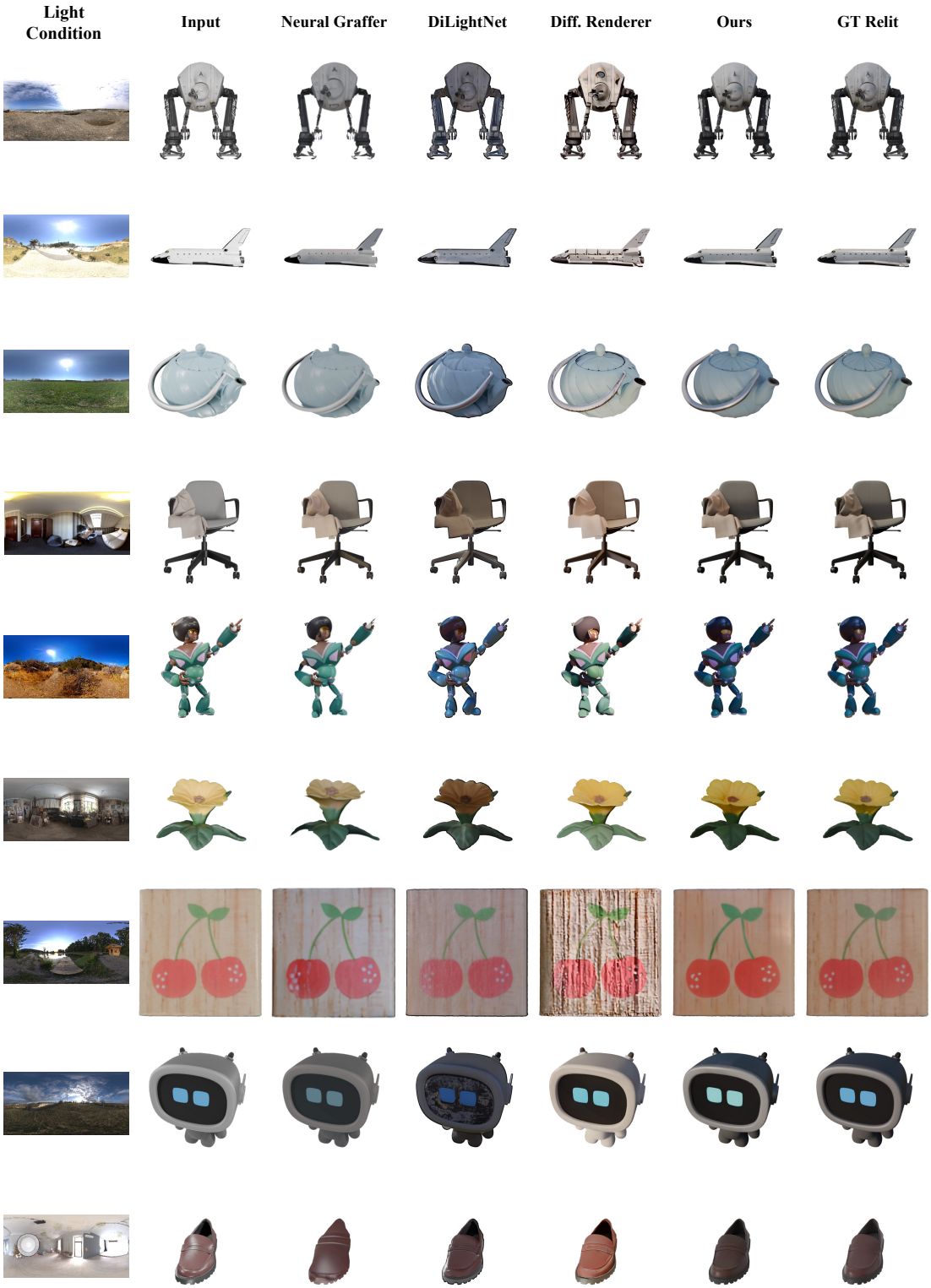

Figure 16. More visualization results of relighting and rendering from a single image under unknown illumination.

Figure 17. More visualization results of relighting and rendering from a single image under unknown illumination.

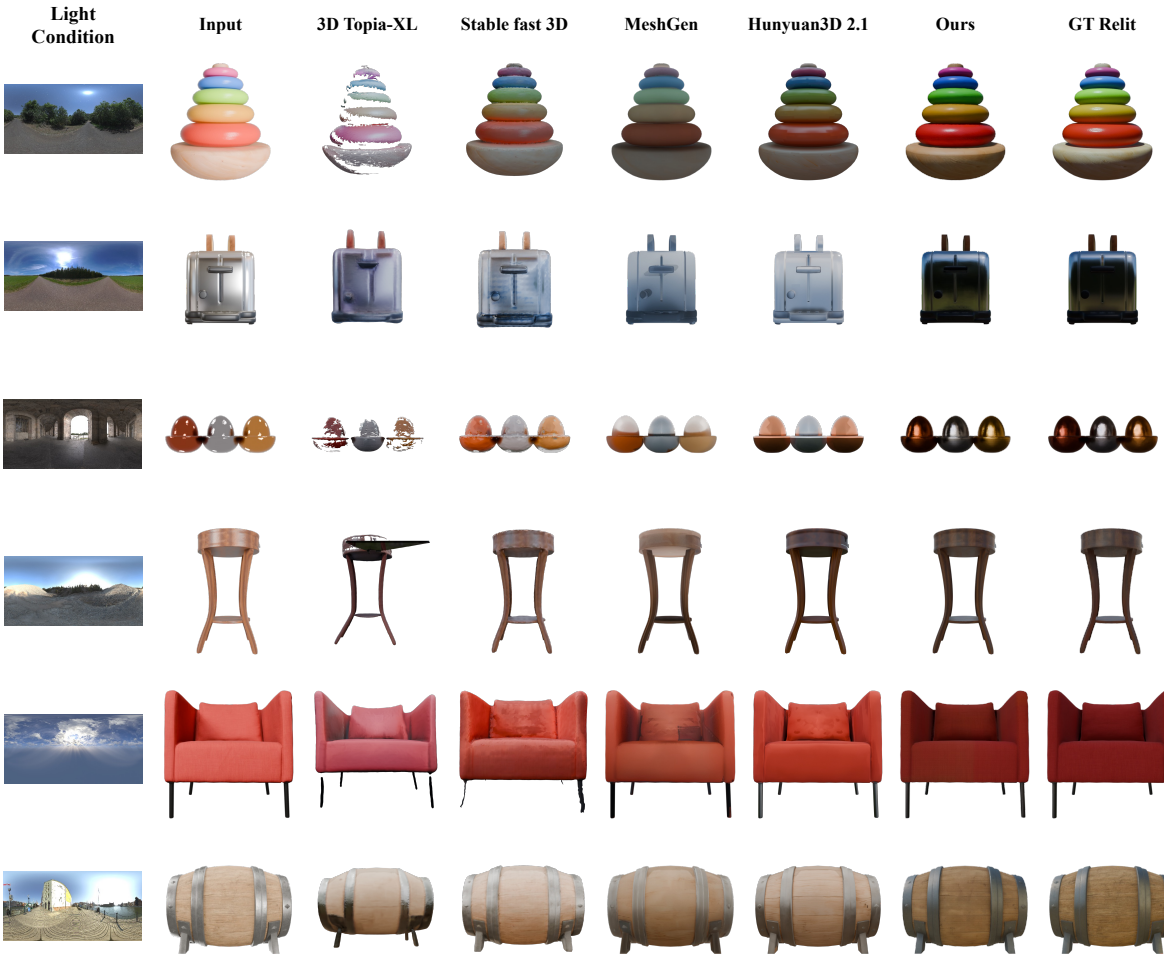

Figure 18. Comparison of relighting renderings between our neural rendering method and 3D generation methods that can recover PBR material properties. Our method achieves more stable and accurate rendering results.

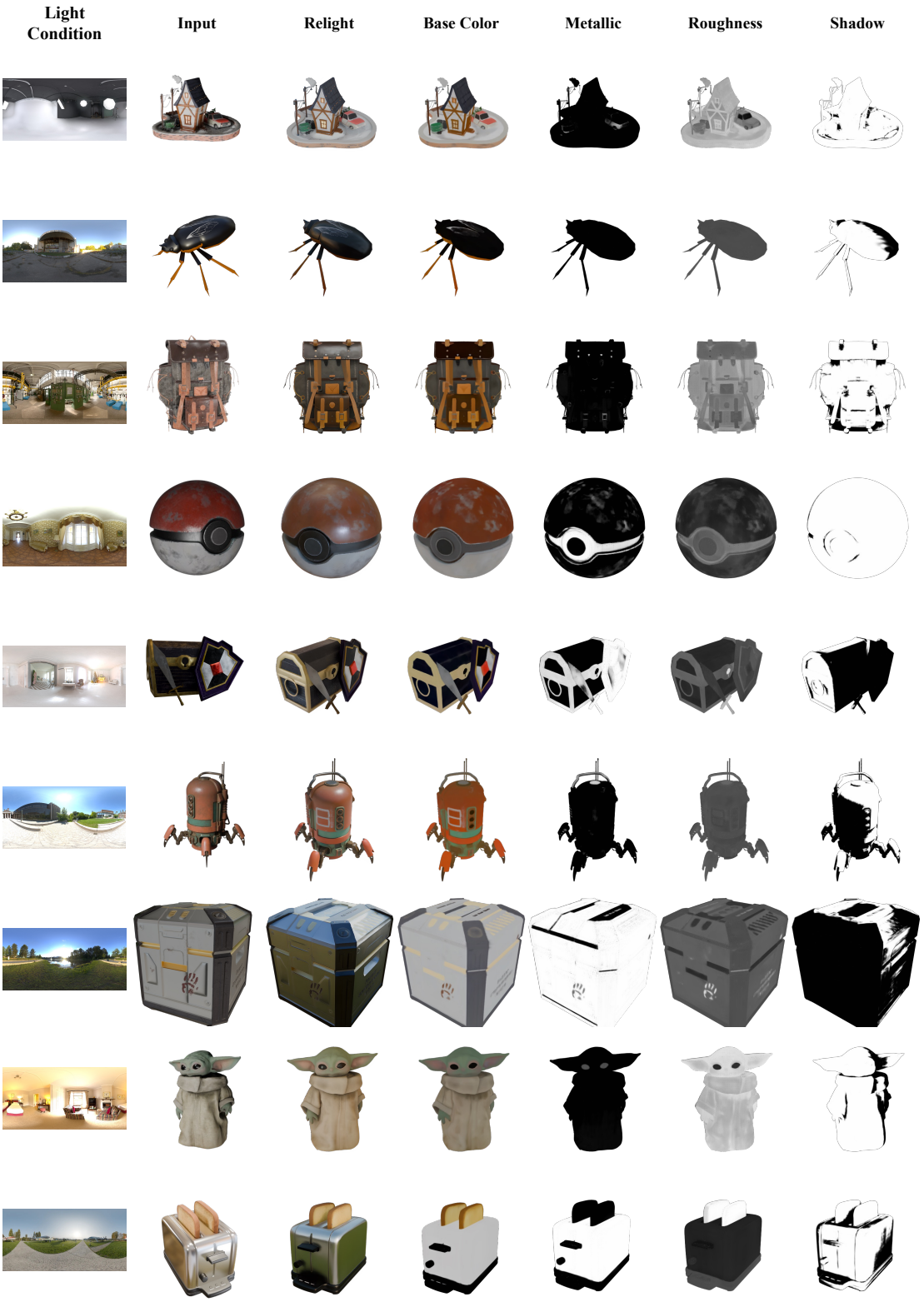

Figure 19. Additional relighting results from a single image under target illumination, along with the PBR materials and shadows estimated by our method.