PBR-NeRF: Inverse Rendering with Physics-Based Neural Fields

Supplementary Material

A. Ablation Study Qualitative Results

We present qualitative results in Fig. 6 for the ablation study of the proposed physics-based losses on the NeILF++ dataset City scene in Tab. 3 of Sec. 4.3. To ensure a fair comparison, all ablated versions share the same hyperparameters as our proposed method described in Sec. 4.1, varying only the inclusion of the proposed loss terms.

The baseline (ID 1) of NeILF++ [51] without the Lambertian loss, exhibits many "baked-in" specular highlights. As shown in Fig. 6, the baseline's predicted albedo contains many specular artifacts being added to the diffuse lobe. Specifically, there is fringing on the sphere, shadowing artifacts on the cube (particularly on its lower regions shadowed by the other two objects), and an overly bright floor due to the entanglement of the diffuse and specular lobes. These specular effects in the albedo cause incorrect patches on the shadowed cube in the metallicness prediction and overly dark patches in the bottom left of the incident light estimation for the associated cube location.

Adding the Conservation of Energy loss (ID 2) mitigates these artifacts by enforcing energy conservation for the diffuse and specular lobes. As a result, the lighting estimation is also improved since the estimated light does not need to compensate for the previously non-energy-conserving (overly reflective) predicted materials. The predicted lighting is more consistent and improves the originally overly dim patches in the bottom left corner of the baseline (corresponding to the cube). Reducing specular artifacts in the lighting also improves the material estimation, as shown by the increased albedo, roughness, and metallicness PSNR in Tab. 3. Qualitatively, we also see that the albedo of the helmet and cube are effectively recovered in all regions except the shadowed area. Despite these improvements, there are still fringing artifacts on the sphere as well as shadowing effects on the ground and cube.

Adding the specular loss instead of the Conservation of Energy loss (ID 3), shows similar improvements in albedo, roughness, and metallicness over the baseline (ID 1).

Finally, incorporating both the Conservation of Energy and specular losses in our final method (ID 4) improves the albedo, roughness, and metallicness even further. We once again highlight that using both our novel physics-based losses improves roughness by 0.35 PSNR, metallicness by 0.27 PSNR, and albedo by 3.28 PSNR over the baseline (ID 1). Qualitatively, the albedo estimation, which is most important, is significantly improved in our final method (ID 4), especially with the green albedo of the helmet and the gray albedo of the cube. The roughness and metallicness of the shadowed area of the cube are also recovered with fewer artifacts. However, there are still limitations with our proposed method, which highlights directions for future improvement. In particular, we fail to fully reproduce the specular effects in the helmet's reflective visor and metallic pieces in the predicted RGB. This limitation is likely due to our overestimation of roughness and the missing fine details in our metallicness prediction. Furthermore, we still exhibit limited "baked-in" specular effects in our predicted albedo with fringing on the sphere and incorrectly predict shadowy patches on the cube and floor.

Overall, we convincingly improve the state-of-the-art material estimation over our baseline NeILF++ [51]. While challenges remain in handling high-frequency specular effects and reflective surfaces, our method convincingly advances the state of the art in joint material and lighting estimation.

B. Additional Qualitative Results

We present additional qualitative results for the *Env* lighting subset of the NeILF++ dataset [51] in Fig. 7 for the corresponding quantitative results in Tab. 1.

Using our physics-based losses, our method improves both lighting and material estimation compared to the NeILF++ baseline. These improvements are consistent with the trends observed in the more challenging *Mix* subset shown in Fig. 4 of the main paper. For the *Env* subset, we significantly reduce artifacts in the upper halves of the estimated incident light fields and enhance albedo estimation in most scenes.

Notably, in the Studio scene, while some specular patchy effects persist on the helmet, cube, sphere, and floor, our method better captures finer details in the helmet's metallicness and roughness, which are absent in the NeILF++ baseline. Across the *Env* subset, our roughness and metallicness predictions show considerable improvements, including the removal of shadowy patches on the cube and fringe artifacts on the sphere.

Although some limitations remain, such as missing fine-grained details on the helmet, our physics-based losses deliver consistent gains in material and lighting estimation, highlighting their effectiveness across diverse lighting conditions.

C. Non-Energy Conserving BRDFs

We extend the discussion in Section 3.2 and provide additional intuition for when the Conservation of Energy property for BRDFs fails to hold. The Disney BRDF cre-

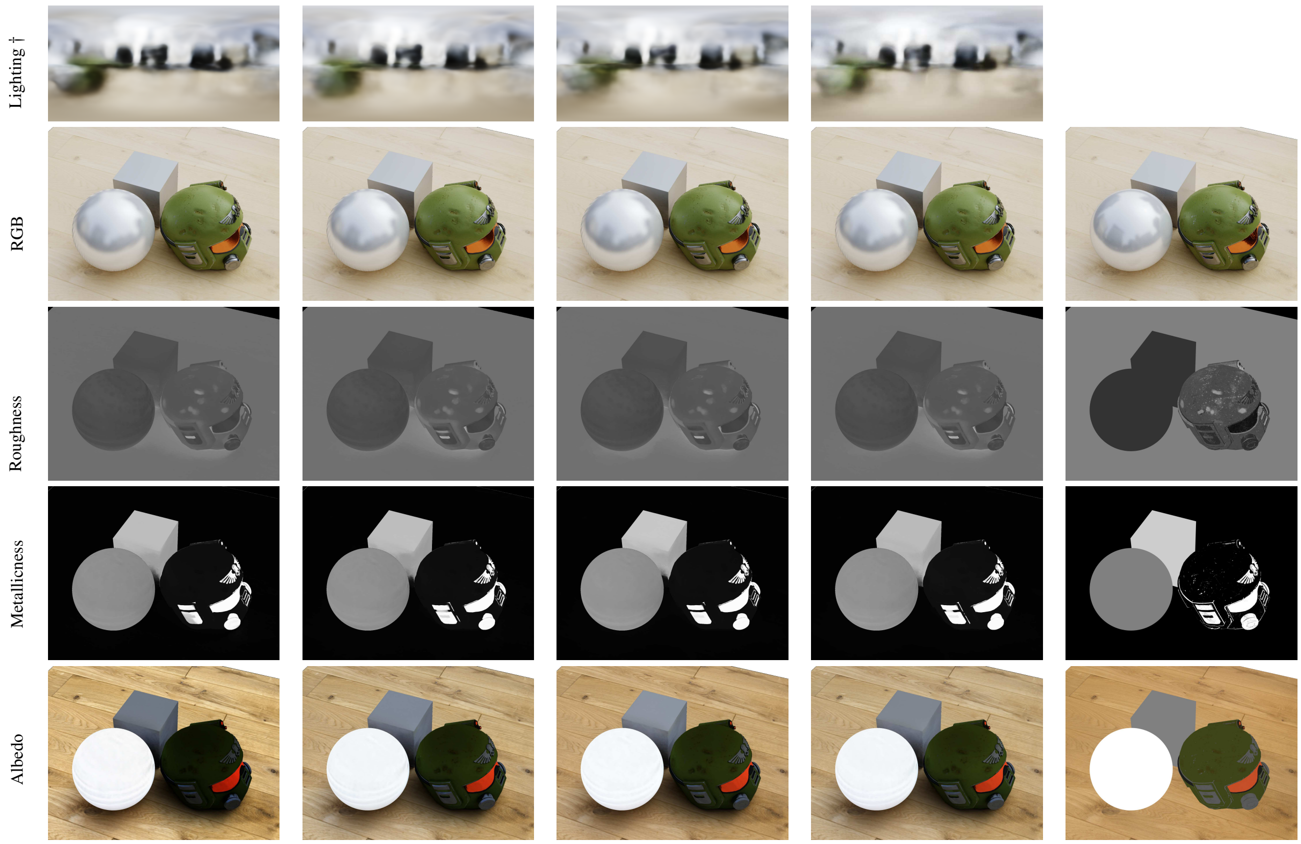

Figure 6. **Ablation of our physics-based losses.** Qualitative performance of ablated versions of PBR-NeRF and of its full version is compared on the *City* illumination with predicted illumination and Disney BRDF parameter estimation on novel views. †: no ground-truth environment maps are provided with the dataset.

ates excess energy for unnormalized terms or when the Schlick Fresnel approximation is invalid, causing overly bright specular highlights [10]. Further, energy can be lost as the underlying microfacet BRDF models rough materials with *single* reflection instead of multiple scattering on the microsurface.

D. Additional Background Details

For completeness, we provide further implementation details on the NeILF++ implicit differential renderer (Sec. D.1), joint material-illumination-geometry optimization (Sec. D.2), and hyperparameter sweeps (Sec. D.3).

D.1. Implicit Differential Renderer (IDR)

The second term of the specular BRDF f_s in Equation (4) is the Fresnel term $F(\boldsymbol{\omega}_o, \boldsymbol{\omega}_h) \in \mathbb{R}^3$. It models glossy reflection at glancing angles due to Fresnel reflection. Following [11], we use the Schlick Fresnel Approximation

$$F(\boldsymbol{\omega}_o, \boldsymbol{\omega}_h; b, m) = F_0 + (1 - F_0)(1 - \boldsymbol{\omega}_o \cdot \boldsymbol{\omega}_h)^5 \quad (10)$$

where

$$F_0 = 0.04(1 - m) + bm \quad (11)$$

The third term in Equation (4) is the geometry term $G(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o, \mathbf{n}; r) \in \mathbb{R}$. It models the masking and shadowing of microfacets depending on the incident and viewing direction.

$$G(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o, \mathbf{n}; r) = G_1(\boldsymbol{\omega}_i \cdot \mathbf{n}) G_1(\boldsymbol{\omega}_o \cdot \mathbf{n}) \quad (12)$$

where

$$G_{\text{GGX}}(z; r) = \frac{2}{(2 - r^2)z + r^2} \quad (13)$$

as implemented in NeILF++ [51].

D.2. Joint Optimization

We provide a more detailed description of the joint material-illumination-geometry optimization discussed in Sec. 3.3. We summarize all 3 PBR-NeRF training phases and their losses in Tab. 4. The final weightings used for each loss function term are listed in Tab. 5.

Table 4. **Summary of PBR-NeRF geometry, material, and joint training phases** with their respective losses. ↓: loss is downscaled from the previous stage. ■: the point cloud loss is optionally used to improve geometry SDF initialization. †: loss used when no ground-truth geometry is provided (e.g. DTU dataset). $*$: our novel PBR losses, namely (1) the Conservation of Energy Loss $\mathcal{L}_{\text{cons}}$ and (2) the NDF-weighted Specular Loss $\mathcal{L}_{\text{spec}}$.

	Geometry-based Losses					Material-based Losses				
Optimization Phase	$\mathcal{L}_{\text{NeRF}}$	$\mathcal{L}_{\text{pcd}}$ †	$\mathcal{L}_{\text{Eik}}$ †	$\mathcal{L}_{\text{Hess}}$ †	$\mathcal{L}_{\text{surf}}$ †	$\mathcal{L}_{\text{pbr}}$	$\mathcal{L}_{\text{ref}}$	$\mathcal{L}_{\text{smth}}$	$\mathcal{L}^*_{\text{cons}}$	$\mathcal{L}^*_{\text{spec}}$
Geometry	✓	■	✓	✓	✓	✗	✗	✗	✗	✗
Material	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓
Joint	✓	✗	✓	↓	↓	✓	✓	✓	✓	✓

Table 5. **PBR-NeRF loss function term weighting** for the NeILF++ and DTU datasets. †: the point cloud loss $\mathcal{L}_{\text{pcd}}$ is only used in the geometry phase for NeRF-SDF initialization. $*$: our novel PBR losses, namely (1) the Conservation of Energy Loss $\mathcal{L}_{\text{cons}}$ and (2) the NDF-weighted Specular Loss $\mathcal{L}_{\text{spec}}$.

	Geometry-based Losses					Material-based Losses				
	λ_{NeRF}	λ_{pcd} †	λ_{Eik}	λ_{Hess}	λ_{surf}	λ_{pbr}	λ_{ref}	λ_{smth}	λ^*_{cons}	λ^*_{spec}
NeILF++ [51]	1.0	N/A	N/A	N/A	N/A	1.0	0.1	0.0005	0.01	0.5
DTU [15]	1.0	0.1	0.1	0.001	0.01	1.0	0.1	0.0005	0.01	0.01

Geometry phase. During the geometry phase, the NeRF SDF network is learned using the following loss.

$$\mathcal{L}_{\text{geo}} = \lambda_{\text{NeRF}}\mathcal{L}_{\text{NeRF}} + \lambda_{\text{pcd}}\mathcal{L}_{\text{pcd}} + \lambda_{\text{Eik}}\mathcal{L}_{\text{Eik}} + \lambda_{\text{Hess}}\mathcal{L}_{\text{Hess}} + \lambda_{\text{surf}}\mathcal{L}_{\text{surf}} \tag{14}$$

The NeRF rendering loss $\mathcal{L}_{\text{NeRF}}$ is identical to the original NeRF formulation. The estimated RGB color $L_{o,\text{NeRF}}$ is rendered using the NeRF network and compared to the ground truth color c using $\mathcal{L}_{\text{NeRF}}$ with a mean squared error.

$$\mathcal{L}_{\text{NeRF}} = ||L_{o,\text{NeRF}} - c||_2^2 \tag{15}$$

When ground truth geometry is provided, as in the NeILF++ dataset [51], the estimated RGB color $L_{o,\text{NeRF}}$ is simply evaluated at the ground truth surface point using the viewing direction instead of using volume rendering. Since we have the ground truth geometry, the other geometry priors are not needed and their corresponding losses are assigned a zero weight. Therefore, only the NeRF rendering loss $\mathcal{L}_{\text{NeRF}}$ is used, with the only nonzero loss weight being λ_{NeRF}.

When ground truth geometry is not provided, we use additional geometry priors encoded as loss function terms.

Following [50, 51], we optimize the SDF with a point cloud loss only during the geometry phase. The point cloud loss supervises the predicted SDF distances and normals:

$$\mathcal{L}_{\text{pcd}} = |\mathbb{G}(\mathbf{x}_{\text{pcd}})| + \left(1 - \mathbf{n}_{\text{pcd}} \cdot \frac{\nabla_{\mathbf{x}}\mathbb{G}(\mathbf{x}_{\text{pcd}})}{||\nabla_{\mathbf{x}}\mathbb{G}(\mathbf{x}_{\text{pcd}})||}\right). \tag{16}$$

where $\mathbf{x}_{\text{pcd}}$ is a point in the point cloud, $\mathbb{G}(\mathbf{x}_{\text{pcd}})$ represents the SDF's predicted signed distance at $\mathbf{x}_{\text{pcd}}$, and $\mathbf{n}_{\text{pcd}}$ is the point cloud normal at $\mathbf{x}_{\text{pcd}}$. This point cloud loss is only used to help initialize the SDF to a satisfactory quality and we perform full material-illumination-geometry optimization without point cloud input during the later joint phase. This use of point clouds is similar to the use of point clouds in 3D Gaussian Splatting for initializing 3D Gaussians.

Note that the point cloud loss $\mathcal{L}_{\text{pcd}}$ is strictly optional, as indicated in Tab. 4. Following NeILF++ [51], we use the point cloud loss $\mathcal{L}_{\text{pcd}}$ on the DTU dataset to improve initial SDF quality and for a fair performance comparison.

We additionally use the Eikonal loss to penalize the SDF when the gradient at the surface point $\mathbb{G}(\mathbf{x})$ does not have a magnitude of 1.

$$\mathcal{L}_{\text{Eik}} = \big|||\nabla_{\mathbf{x}}\mathbb{G}(\mathbf{x})|| - 1\big| \tag{17}$$

Furthermore, the Hessian loss penalizes rapidly changing gradient directions by minimizing the Hessian matrix norm.

$$\mathcal{L}_{\text{Hess}} = ||\mathbf{H}\mathbb{G}(\mathbf{x})||_1 \tag{18}$$

Finally, the minimal surface loss encourages compact interpolation and extrapolation of unobserved surfaces by minimizing the surface elastic energy.

$$\mathcal{L}_{\text{surf}} = \delta_\epsilon(\mathbb{G}(\mathbf{x}))) \tag{19}$$

where δ_e is the regularized Dirac delta function parametrized by a sharpness ϵ,

$$\delta_\epsilon(z) = \frac{\epsilon\pi^{-1}}{\epsilon^2 + z^2} \tag{20}$$

Material Phase. During the material phase, we train the NeILF and BRDF MLPs with the frozen NeRF SDF weights using the material phase loss:

$$\mathcal{L}_{\text{mat}} = \lambda_{\text{pbr}}\mathcal{L}_{\text{pbr}} + \lambda_{\text{smth}}\mathcal{L}_{\text{smth}} + \lambda_{\text{ref}}\mathcal{L}_{\text{ref}} + \lambda_{\text{cons}}\mathcal{L}_{\text{cons}} + \lambda_{\text{spec}}\mathcal{L}_{\text{spec}} \quad (21)$$

The physically based rendering loss $\mathcal{L}_{\text{pbr}}$ supervises the estimated outgoing radiance from (6):

$$\mathcal{L}_{\text{pbr}} = ||L_{o,\text{NeILF++}} - c||_2^2 \quad (22)$$

Following [44, 51], we use a bilateral smoothness loss $\mathcal{L}_{\text{smth}}$ to encode the assumption that roughness r and metallicness m at surface point x_p are smooth if the color of corresponding pixel p has no sharp gradients:

$$\mathcal{L}_{\text{smth}} = (||\nabla_{\mathbf{x}} r(\mathbf{x}_p)|| + ||\nabla_{\mathbf{x}} m(\mathbf{x}_p)||)\exp(-||\nabla_p I_p||) \quad (23)$$

where $\nabla_p I_p$ is the image gradient at pixel p.

We also use the NeILF++ inter-reflection loss to use the NeRF SDF predicted outgoing radiance to supervise the incident light predicted by the NeILF MLP along the same ray $\boldsymbol{\omega}_i = -\boldsymbol{\omega}_o$ between two surface points $\mathbf{x}_1$ and $\mathbf{x}_2$.

$$\mathcal{L}_{\text{ref}} = ||L_{i,\text{NeILF}}(\mathbf{x}_2, \boldsymbol{\omega}_i) - L_{o,\text{NeRF}}(\mathbf{x}_1, -\boldsymbol{\omega}_i)||_1 \quad (24)$$

Joint Phase. During the joint optimization phase, the NeRF SDF, BRDF, and NeILF MLPs have been pre-trained by the previous geometry and material phases, allowing us to jointly optimize all networks simultaneously. We reuse the geometry phase loss $\mathcal{L}_{\text{geo}}$ and material phase loss $\mathcal{L}_{\text{mat}}$ from (14) and (21), respectively, to obtain the overall joint phase loss

$$\mathcal{L}_{\text{joint}} = \mathcal{L}_{\text{geo}} + \mathcal{L}_{\text{mat}} \quad (25)$$

Note that the NeRF SDF is used twice per training sample during the Joint phase: (1) estimating $L_{\text{o,NeRF}}$ for $\mathcal{L}_{\text{geo}}$ and $\mathcal{L}_{\text{ref}}$; (2) sphere tracing to compute $\mathbf{x}$ and $\mathbf{n}$ for $L_{\text{o,PBR}}$.

D.3. Hyperparameter Sweeps

We now specify how we picked various hyperparameters introduced with our novel Conservation of Energy Loss (7) and NDF-weighted Specular Loss (8).

The Conservation of Energy Loss $\mathcal{L}_{\text{cons}}$ does not introduce any new hyperparameters. We increase the number of incident light directions from the default $|S_L| = 128$ in NeILF++ to $|S_L| = 256$ to better enforce energy conservation and separate the diffuse and specular lobes using more samples.

For the NDF-weighted Specular Loss, we use a temperature $T_{\text{spec}} = 1$. We include the softmax temperature T_{spec} in our NDF-weighted Specular Loss definition (7) for full generality and to provide an option to control softmax sharpness.

To determine the loss term weightings for our novel physics-based losses, we perform a grid search with $\lambda_{\text{cons}} \in \{1.5, 1.0, 0.5, 0.1, 0.05, 0.01, 0.005\}$ and $\lambda_{\text{spec}} \in \{1.5, 1.0, 0.5, 0.1, 0.05, 0.01, 0.005\}$.

We also remove the Lambertian loss $\mathcal{L}_{\text{Lam}}$ used in NeILF++ by choosing $\lambda_{\text{Lam}} = 0$ to remove the strong limitations on material estimation that the Lambertian assumption causes.

The remaining hyperparameters are identical to NeILF++ [51].

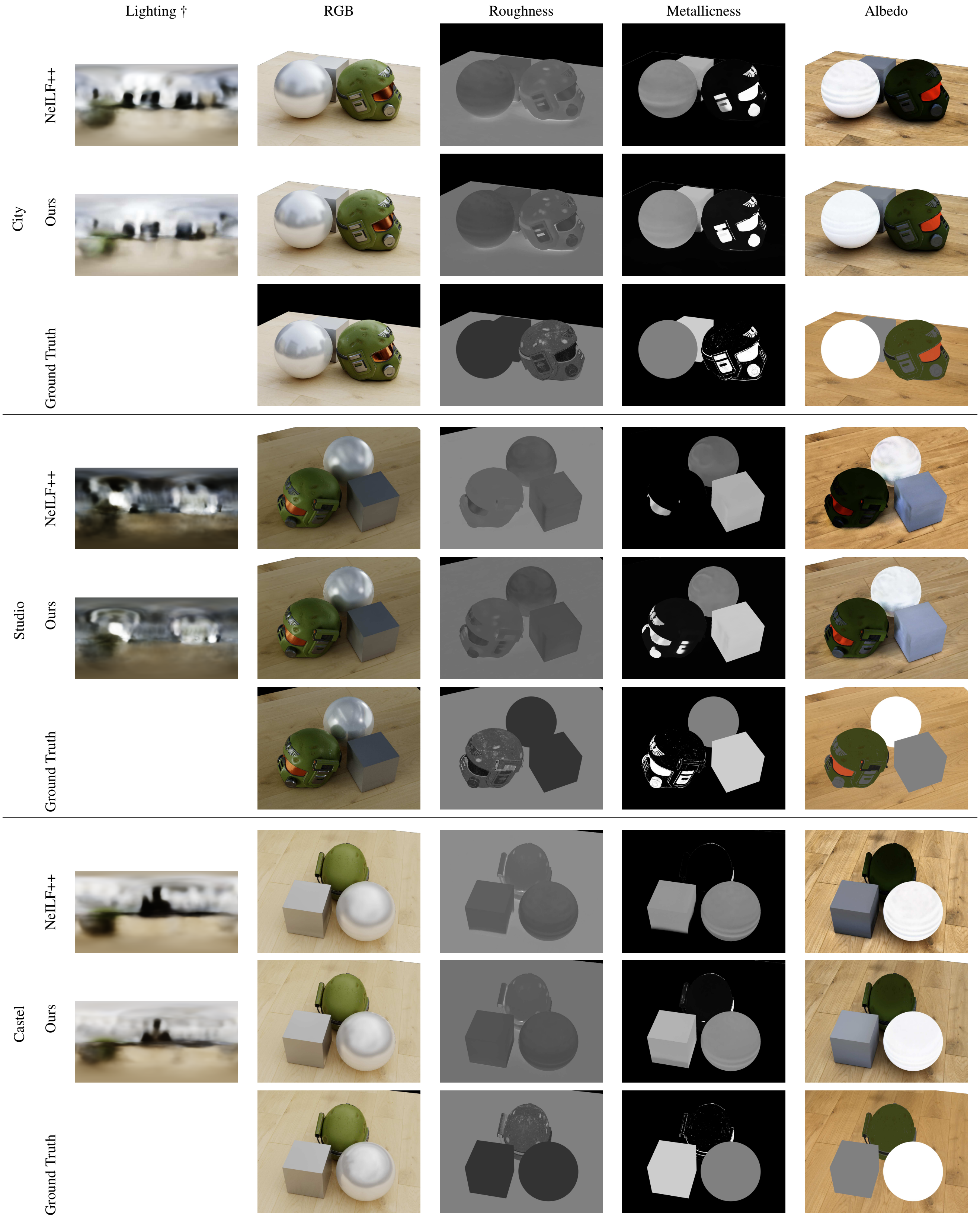

Figure 7. **Additional qualitative comparisons on the NeILF++ dataset [51].** †: no ground-truth environment maps are provided with the dataset.