EXTRACTING NEURAL MATERIALS FROM IMAGES

Anonymous authors
Paper under double-blind review



Figure 1: **NeuMatEx** extracts *Neural Materials*, a richer material representation that goes beyond PBR, from multi-view images. NeuMatEx decomposes a diffuse lobe and neural specular lobes to model complex material-light interactions such as *haze*, *dust*, *clearcoat*, *fuzz*, *scatter*, and their mixtures. The extracted neural materials support real-time path tracing on a commodity GPU.

ABSTRACT

Neural materials can represent complex specular reflections and scattering effects in a compact, universal basis. However, acquiring and authoring such materials remains challenging. We present NeuMatEx, a differentiable inverse rendering method for extracting spatially varying neural materials from images. The nonlinear structure of neural material latent spaces makes optimization with naïve inverse rendering infeasible. To address this, we train a Large Material Reconstruction Model (LMRM) that directly predicts initial base color, neural material latents, and aleatoric uncertainty guides from images. This material prior provides a good initialization and better constrains our subsequent optimization using inverse path tracing. The predicted uncertainty further helps by anchoring high-confidence regions more tightly to the LMRM prediction, preventing lighting and complex specular effects from being baked into materials. Experiments on synthetic and real assets show that NeuMatEx extracts complex materials with better visual quality and material decomposition than PBR-based methods.

1 INTRODUCTION

Authoring materials for 3D objects is a challenging artistic task, which involves creating spatially-varying details and assigning appropriate material parameters to their different parts. Current interactive applications, *e.g.*, games, commonly apply a physically-based (PBR) material model, where a material is defined by diffuse base color, roughness, and metallicity stored in spatially varying textures. This allows a reasonable variety of materials, including metals, woods, and plastics, but is still far from the appearance quality in offline rendering, *e.g.*, film production, where artists design large material node graphs, *e.g.*, MaterialX (MaterialX Project, 2026), with multiple layers to capture the real-world appearance of effects like skin, eyes, and car paint.

To bridge the gap between offline and real-time material quality, *neural* material representations (Kuznetsov et al., 2021; Fan et al., 2022; Zeltner et al., 2024; Yu et al., 2026) have recently

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

been proposed. The idea is to bake complex material graphs into a compact neural representation, consisting of a set of latent texture maps and a small neural network. This results in a unified format efficiently executable on modern graphics hardware at real-time rates, *e.g.*, inside a path tracing real-time renderer (Zeltner et al., 2024). Neural materials are typically baked from large material node graphs by densely sampling the 6D space of spatial variation (2D) and incoming and outgoing directions (4D). For each sample, the spatially varying bidirectional reflection/scattering function (SVBSDF) is evaluated and the neural material is optimized to match the reference material's SVBSDF as closely as possible.

In this work, we ask whether the inverse rendering machinery can be extended to extract *neural material* representations, which offer far greater expressiveness. To help asset creation, we introduce the first pipeline to extract neural materials from multi-view images. Recent works have successfully extracted PBR materials from multi-view images via inverse rendering, leveraging the highly constrained PBR representation. Neural materials, on the other hand, are much more expressive, with additional degrees of freedom, and the latent space representing the manifold of valid materials can be fractured. This makes inverse rendering a highly non-convex problem with many poor local minima, where the optimization can succeed only when initialized within the basin of attraction of a good solution. We address this problem by combining feed-forward reconstruction model priors with differentiable optimization to robustly extract neural materials from images.

Specifically, we have constructed a large feed-forward material reconstruction model (LMRM) which produces 1) a high-quality initial guess of the neural material from a set of image observations, and 2) uncertainty estimates of the predictions. In a second step, we further refine the material prediction using Monte Carlo-based inverse path tracing, guided by uncertainty. We leverage a recent neural material basis (Yu et al., 2026) which focuses on high-quality specular appearance, including clearcoat, dust, fuzz, and scatter effects (see Fig. 1). Our extracted neural materials can be directly deployed inside a real-time path tracer. We summarize our main contributions as follows:

- We introduce the first image-based neural material extraction for 3D objects, combining a Large Material Reconstruction Model (LMRM) with Test-Time Optimization (TTO), outperforming PBR.
- We train an LMRM to estimate neural materials from multi-view images, providing good initialization for our subsequent TTO with inverse path tracing.
- Our uncertainty-guided material regularization better constrains TTO and prevents baked-in effects.

2 Related Work

Our work targets extracting neural materials from images by combining inverse rendering with a large feed-forward reconstruction model. We briefly review the related fields.

Material extraction from images. A long-standing goal in inverse rendering is to extract spatially varying materials from images of 3D shapes. Several techniques estimate surface radiometric properties from images. Previous works on SVBRDF estimation rely on special viewing configurations, lighting patterns or complex capturing setups (Lensch et al., 2003; Gardner et al., 2003; Ghosh et al., 2009; Guarnera et al., 2016; Weinmann & Klein, 2015; Bi et al., 2020; Boss et al., 2020; Schmitt et al., 2020). More recently, optimization-based methods jointly recover PBR texture maps, environment lighting and geometry from casually captured images (Azinovic et al., 2019; Zhang et al., 2021b; Chen et al., 2021; Munkberg et al., 2022; Boss et al., 2021; Hasselgren et al., 2022; Zhang et al., 2022; Lin et al., 2025) using inverse rendering. Radiance field-based methods factorize scene representations into intrinsic components without relying on explicit geometry (Zhang et al., 2021a; Jin et al., 2023; Jiang et al., 2024; Gao et al., 2024; Liang et al., 2024).

Optimization-based methods are sensitive to ambiguous data and often require well-calibrated observations to reliably disentangle lighting, reflectance, and shape. To overcome this, a plethora of methods rely on neural networks to predict BRDFs from images (Gao et al., 2019; Guo et al., 2020; Li et al., 2020; 2018; Nimier-David et al., 2021; Luan et al., 2021). Additionally, many recent methods use feed-forward or generative approaches to directly predict material intrinsics (Kocsis et al., 2024; Zeng et al., 2024; Litman et al., 2025; Youwang et al., 2024; Munkberg et al., 2025; Hasselgren et al., 2026; Kocsis et al., 2026; Li et al., 2025; Siddiqui et al., 2024; Tencent, 2025; Xiang et al., 2026; Li et al., 2026). While the field has advanced substantially, these methods universally

target standard PBR material representations, *i.e.*, a diffuse term paired with a single-lobe specular component, which limits the complexity of the extracted materials.

Neural materials. Neural material models represent SVBSDFs using latent textures for spatial variation and neural decoders for directional variation (Kuznetsov et al., 2021; Fan et al., 2022; Zeltner et al., 2024). To improve rendering performance, the decoders are highly optimized MLPs designed around hardware constraints (Zeltner et al., 2024), but the complexity of the compressed latent space typically requires special consideration during optimization (Bitterli et al., 2026).

Recent work explores neural-material generation through diffusion-based synthesis of neural textures (Raghavan et al., 2025) or reconstruction from generative video priors (Xue et al., 2026). VideoNeuMat (Xue et al., 2026) focuses on obtaining planar 2D neural materials from generated videos, rather than recovering materials on 3D objects from images or real-world photos. In contrast, we combine object-level inverse rendering with the richer neural material basis of Yu et al. (2026). Since the decoder maps latent textures to appearance nonlinearly, test-time optimization is sensitive to initialization. We therefore learn a large-scale image-to-neural-material reconstruction model for initialization and refine its prediction using differentiable path tracing.

Relightable neural representations. Several works use neural representations for relighting surface and volumetric assets (Mullia et al., 2024; Zeng et al., 2023; Rainer et al., 2022; Fan et al., 2025; Xu et al., 2023). They typically treat assets as atomic units and precompute light transport, which makes animation, local lighting, and object interaction difficult. For instance, reconstructed objects generally do not support mutual global illumination effects. In addition, asset generation often requires controlled-lighting setups (Mullia et al., 2024; Fan et al., 2025; Xu et al., 2023), typically with point light sources, complicating real-world data capture. In contrast, our method relies on inverse rendering (Munkberg et al., 2022; Boss et al., 2021; Hasselgren et al., 2022; Zhang et al., 2022) and explicitly reconstructs the local SVBSDF, making the resulting assets compatible with off-the-shelf animation and light-transport algorithms.

3 Overview of NeuMatEx

Neural Materials (NM). Unlike standard PBR materials (Burley, 2012; Karis, 2013) parameterized by explicit maps (base color, roughness, metallicity), a Neural Material (NM) (Kuznetsov et al., 2021; Fan et al., 2022; Zeltner et al., 2024) represents a richer SVBSDF with a set of latent textures and a small neural network. We adopt the recent neural material SVBSDF of Yu et al. (2026), which combines a Lambertian diffuse lobe with an expressive *neural* specular lobe, evaluated at a surface point $\mathbf{p}$ with incoming and outgoing directions ω_i, ω_o:

$$f(\mathbf{p}, \omega_i, \omega_o) = T_{\text{neu}}(\mathbf{p}, \omega_i)\,\frac{\rho_d(\mathbf{p})}{\pi} + f_{\text{neu}}(\mathbf{p}, \omega_i, \omega_o), \tag{1}$$

where ρ_d is the diffuse base color, f_{neu} the specular BSDF, and T_{neu} the transmission albedo enforcing energy conservation. The latter two are decoded by a learned universal decoder MLP $\mathcal{D}_{\text{neu}}$:

$$(f_{\text{neu}}, T_{\text{neu}}, R_{\text{neu}}) = \mathcal{D}_{\text{neu}}(\ell(\mathbf{p}),\, \omega_i, \omega_o), \tag{2}$$

where $\ell(\mathbf{p}) \in \mathbb{R}^6$ is a per-point neural specular latent code that compresses 22 analytical SVBSDF parameters (Yu et al., 2026), and R_{neu} is a scalar measure of reflected energy of f_{neu}, used to importance-sample between the diffuse and neural specular lobes during rendering. Learned from a large-scale 3D shape dataset with procedurally enhanced NM textures (Yu et al., 2026) spanning complex multi-lobe specular effects, *e.g.*, clearcoat and scattering, this NM latent space covers a far richer appearance than PBR's single GGX lobe.

In this work, we aim to extract $\rho_d(\mathbf{p})$ and $\ell(\mathbf{p})$ from images. Since $\ell(\mathbf{p})$ is a neural latent observed through the nonlinear decoder $\mathcal{D}_{\text{neu}}$, not a physical parameter, extracting it is highly non-convex; we thus predict both with a large feed-forward model and refine them via test-time optimization.

NeuMatEx Pipeline. NeuMatEx consists of two stages (see Fig. 2). In the first stage (Fig. 2(a)), a Large Material Reconstruction Model (LMRM) predicts a feature triplane from multi-view images and mesh geometry, which is decoded into an initial neural material and per-point uncertainty estimates. In the second stage (Fig. 2(b)), we refine the triplane via differentiable inverse path tracing, using the predicted uncertainty to regularize the ill-posed material decomposition.

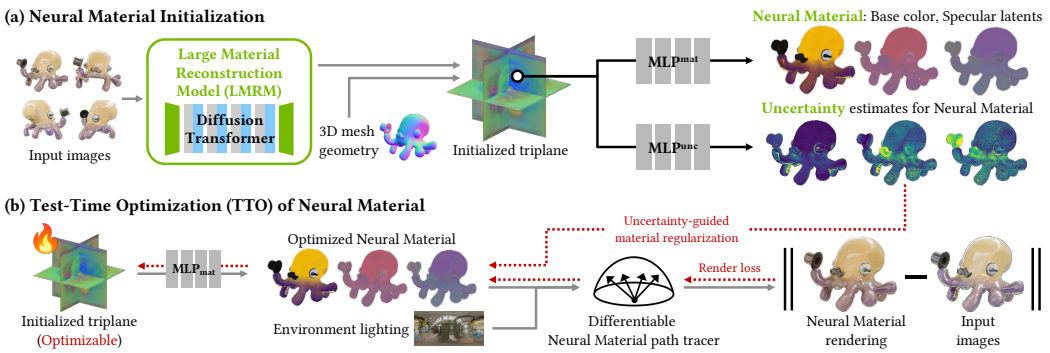

Figure 2: NeuMatEx consists of two stages. **(a) Neural Material Initialization:** Given input images, a Large Material Reconstruction Model (LMRM) predicts a feature triplane in a single forward pass. The triplane is decoded by two MLPs to jointly predict an initial neural material and per-material uncertainty that reflects ambiguous surface regions. **(b) Test-Time Optimization (TTO):** We further optimize the initialized triplane via differentiable path tracing supervised by a photometric render loss against the input images. To regularize the ill-posed inverse rendering, uncertainty estimates from (a) are used to guide the TTO, allowing flexible material drifts for uncertain regions and preventing significant material drifts for confident regions, steering the solution away from local minima.

4 Large Material Reconstruction Model

We introduce a large feed-forward material reconstruction model (LMRM) that takes multi-view images and jointly predicts 1) a high-quality initial guess of the neural material, and 2) uncertainty estimates of the material predictions.

Model architecture. Following recent work (Wei et al., 2024; Zhang et al., 2025a; Xue et al., 2026), we leverage a pre-trained text-to-video diffusion transformer (DiT) (Peebles & Xie, 2023), Wan2.1-1.3B (Team Wan, 2025), repurposed as a *single-step* model. We fine-tune the model to take a set of input views, $\mathbf{I}$, and to output *neural materials* parameterized in a triplane representation. We drop the text conditioning. The model comprises a VAE encoder-decoder pair, $(\mathcal{E}_{\text{VAE}}, \mathcal{D}_{\text{VAE}})$, and a transformer-based denoising function, $\mathcal{F}_\theta$. We use the encoder $\mathcal{E}_{\text{VAE}}$ to encode the input image views, $\mathbf{I}$, into a latent tensor, $\mathbf{z}^{\mathbf{I}}$. Specifically, we encode $F = 17$ views, evenly spaced in a 360-degree orbit around the object, alongside the six canonical views. The 17 views are encoded using the video VAE, and the canonical views are each encoded with the image VAE for additional quality (Wang et al., 2024). We then run a single step of the denoising function, $\mathcal{F}_\theta$, to produce an output latent, $\mathbf{z}^{\mathbf{tri}}(\theta) = \mathcal{F}_\theta(\mathbf{z}^{\mathbf{I}})$. This latent is decoded via the VAE decoder, $\mathcal{D}_{\text{VAE}}$, into three triplane features: $(\mathbf{T}_{\text{XY}}, \mathbf{T}_{\text{YZ}}, \mathbf{T}_{\text{XZ}}) = \mathcal{D}_{\text{VAE}}(\mathbf{z}^{\mathbf{tri}}(\theta))$, representing the XY, YZ, and XZ planes, respectively.

We query the triplanes at each surface point, $\mathbf{p}$, for each covered pixel in each view. These three point feature tensors are then concatenated to form a point feature, which is decoded by two small multi-layered perceptrons (MLPs) sharing the same input: a material decoder, $\mathcal{M}_\phi^{\text{mat}}$, and an uncertainty decoder, $\mathcal{M}_\psi^{\text{unc}}$. By querying triplane features for all the surface points, the material and uncertainty decoders produce the respective material G-buffer, $\mathbf{G}_{\text{mat}}$, and per-material uncertainty (log-variance) G-buffer, $\mathbf{G}_{\text{unc}}$, for each covered pixel as follows:

$$\mathbf{G}_{\text{mat}}(\mathbf{p}) = \mathcal{M}_\phi^{\text{mat}}([\mathbf{T}_{\text{XY}}(\mathbf{p}), \mathbf{T}_{\text{YZ}}(\mathbf{p}), \mathbf{T}_{\text{XZ}}(\mathbf{p})]), \tag{3}$$

$$\mathbf{G}_{\text{unc}}(\mathbf{p}) = \mathcal{M}_\psi^{\text{unc}}([\mathbf{T}_{\text{XY}}(\mathbf{p}), \mathbf{T}_{\text{YZ}}(\mathbf{p}), \mathbf{T}_{\text{XZ}}(\mathbf{p})]). \tag{4}$$

Note that we train over a large collection of 3D shapes, so both MLPs, $\mathcal{M}_\phi^{\text{mat}}$ and $\mathcal{M}_\psi^{\text{unc}}$, represent shared, *universal* decoders of triplane features, while the predicted triplanes are unique per 3D object.

Training LMRM. We jointly train the parameters of the LMRM network, θ, and the decoder MLPs ϕ and ψ, by minimizing the training objective: $\mathcal{L}(\theta, \phi, \psi) = \mathcal{L}_{\text{mat}} + \lambda_{\text{unc}}\mathcal{L}_{\text{unc}}$, which combines a material regression loss and an uncertainty loss. The material loss supervises the mean prediction of $\mathcal{M}_\phi^{\text{mat}}$ against the reference, *i.e.*, ground-truth, G-buffers,

$$\mathcal{L}_{\text{mat}} = \mathbb{E}_{\mathbf{G}_{\text{mat}}^{\text{ref}} \sim p_{\text{data}}} \|\mathbf{G}_{\text{mat}}(\theta, \phi) - \mathbf{G}_{\text{mat}}^{\text{ref}}\|_2^2, \tag{5}$$

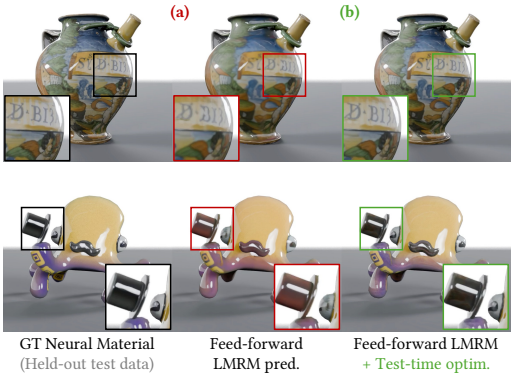

Figure 3: **Why need test-time optimization?** While our feed-forward prediction gives reasonable initial materials (a), test-time optimization (b) recovers finer details and corrects color shifts and material decomposition in challenging cases. All results relit with novel light probe.

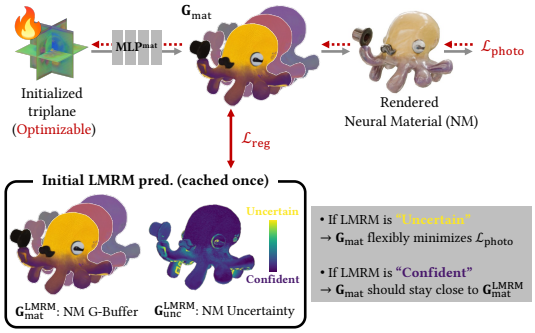

Figure 4: **Uncertainty-guided test-time optimization.** We optimize the LMRM-initialized triplane, decoding it into neural materials and rendering via Monte Carlo path tracing (*top*). An uncertainty-based regularization anchors the solution to the initialization in low-uncertainty regions (*bottom*).

where $\mathbf{G}_{\text{mat}}^{\text{ref}}$ denotes G-buffer views of the reference material parameters. The uncertainty loss supervises the per-pixel material uncertainties following the β-NLL formulation (Seitzer et al., 2022):

$$\mathcal{L}_{\text{unc}} = \mathbb{E}\left[\lfloor \exp\left(\beta \mathbf{G}_{\text{unc}}\right) \rfloor \left(\frac{\mathbf{G}_{\text{unc}}}{2} + \frac{\| \lfloor \mathbf{G}_{\text{mat}} \rfloor - \mathbf{G}_{\text{mat}}^{\text{ref}} \|_2^2}{2 \exp(\mathbf{G}_{\text{unc}})} \right)\right], \tag{6}$$

where $\beta = 0.5$ and $\lfloor \cdot \rfloor$ denotes the stop-gradient operation, applied per material channel. We apply the stop-gradient to the material mean $\mathbf{G}_{\text{mat}}$ inside $\mathcal{L}_{\text{unc}}$ so that it trains only the uncertainty decoder $\mathcal{M}_{\psi}^{\text{unc}}$, while $\mathcal{L}_{\text{mat}}$ trains the mean; please refer to Seitzer et al. (2022) for details.

We adopt a two-stage training curriculum. We first pre-train LMRM on large-scale 3D shape datasets with annotated PBR materials (Deitke et al., 2023; Litman et al., 2025; Zhang et al., 2025b), where $\mathcal{M}_{\phi}^{\text{mat}}$ predicts the standard PBR G-buffers. This PBR pre-training stage lets the model learn a robust image-to-material mapping before facing the more expressive neural material target. We then fine-tune LMRM on 3D assets annotated with procedurally enhanced neural materials (Yu et al., 2026), where $\mathcal{M}_{\phi}^{\text{mat}}$ predicts the neural material G-buffer, *i.e.*, diffuse base color ρ_d and the specular latent code ℓ, exposing $\mathcal{M}_{\psi}^{\text{unc}}$ to the increased ambiguity of recovering multi-lobe reflectance.

5 Test-Time Optimization of Neural Materials

Given initial neural material predictions from LMRM, our test-time optimization (TTO) performs differentiable Monte Carlo inverse rendering to further refine material details and improve decomposition (see Fig. 3). Our MC inverse renderer follows NVDiffRecMC (Hasselgren et al., 2022) but with the PBR specular lobe replaced by a universal neural material basis (Yu et al., 2026).

Inverse rendering optimization for NM. In our setup, the neural component representing non-diffuse contributions consists of a universal, frozen decoder MLP $\mathcal{D}_{\text{neu}}$ operating on a 6D latent code $\ell(\mathbf{p})$ (Eq. 2). Rather than optimizing the per-point material attributes directly, we expose them through our triplane representation $\mathbf{T}$ (see Sec. 3). Together with the frozen material decoder $\mathcal{M}_{\phi}^{\text{mat}}$, the triplane yields the per-point base color $\rho_d(\mathbf{p})$ of the traditional diffuse lobe (Hasselgren et al., 2022) and the specular latent $\ell(\mathbf{p})$, *i.e.*, $(\rho_d(\mathbf{p}),\ \ell(\mathbf{p})) = \mathcal{M}_{\phi}^{\text{mat}}(\mathbf{T}(\mathbf{p}))$, so that the LMRM-initialized triplane $\mathbf{T}$ (Sec. 4) forms the only optimization parameter.

For a given camera pose c, our differentiable path tracer produces an image $\mathbf{I}(\mathbf{T}; c)$. We light the scene with a known high dynamic range (HDR) environment map, and the renderer produces an output image with linear HDR values. We emphasize that our neural materials are operating in linear HDR; thus, strong specular peaks can have significant energy. The reference image $\mathbf{I}_{\text{ref}}(c)$ from the



Figure 5: NeuMatEx reconstructs relightable neural materials from images, demonstrating complex multi-lobe effects, *e.g.*, haze, clearcoat, dust, fuzz, scattering. More visual results in Figs. S3 & S4.

same camera is stored in LDR. We apply a tonemap operator $\mathcal{T}$ with a learnable camera response function (CRF) (Lin et al., 2025) that maps the renderer output to LDR. The rendering loss is:

$$\mathcal{L}_{\text{photo}} = \mathbb{E}_c \Big[\| \mathcal{T}\big(\mathbf{I}(\mathbf{T}; c)\big) - \mathbf{I}_{\text{ref}}(c) \|_2^2 \Big]. \tag{7}$$

We minimize the rendering loss using Adam (Kingma & Ba, 2015), with gradients obtained through differentiable path tracing. Note that both the triplane material decoder $\mathcal{M}_\phi^{\text{mat}}$ and the neural material decoder $\mathcal{D}_{\text{neu}}$ remain frozen throughout; only the triplane $\mathbf{T}$ and the CRF parameters are updated.

Uncertainty-guided material regularization. The photometric objective in Eq. 7 is under-constrained at the neural material level: due to the rich multi-lobe specular effects, many combinations of base color ρ_d and specular latent ℓ can explain the same image-space observations. Thus, the test-time optimization may drift toward an implausible material while still minimizing the photometric loss. To mitigate this ambiguity, we leverage the joint material-uncertainty prediction from LMRM (Eq. 4). In a forward pass, LMRM predicts both the initial material mean $\mathbf{G}_{\text{mat}}^{\text{LMRM}}(\mathbf{p})$ and the log-variance $\mathbf{G}_{\text{unc}}^{\text{LMRM}}(\mathbf{p})$. We freeze both as constants during TTO and use them to *anchor* the current material $\mathbf{G}_{\text{mat}}(\mathbf{T})$, through an uncertainty-weighted regularization term:

$$\mathcal{L}_{\text{reg}} = \mathbb{E}_{\mathbf{p}} \left[\frac{\| \mathbf{G}_{\text{mat}}(\mathbf{T}; \mathbf{p}) - \mathbf{G}_{\text{mat}}^{\text{LMRM}}(\mathbf{p}) \|_2^2}{\exp\big(\mathbf{G}_{\text{unc}}^{\text{LMRM}}(\mathbf{p})\big)} \right], \tag{8}$$

where the error is computed per material channel and reciprocally weighted by the predicted uncertainty $\exp(\mathbf{G}_{\text{unc}}^{\text{LMRM}})$, mirroring the β-NLL term used in training (Eq. 6). Intuitively, Eq. 8 acts as a Gaussian prior centered at the LMRM prediction: confident 3D regions with small $\exp(\mathbf{G}_{\text{unc}}^{\text{LMRM}})$ are strongly anchored, suppressing material drift, while uncertain regions with large $\exp(\mathbf{G}_{\text{unc}}^{\text{LMRM}})$ are loosely constrained and free to drift from the feed-forward initialization (see Fig. 4).

Overall, the complete test-time optimization objective combines the photometric data term (Eq. 7) with the uncertainty-guided material regularizer (Eq. 8), *i.e.*, $\arg\min_{\mathbf{T}} \mathcal{L}_{\text{photo}} + \lambda_{\text{reg}} \mathcal{L}_{\text{reg}}$, where λ_{reg} weights uncertainty-guided material regularization for robust material decomposition.

6 Experiments

In our experiments, we visualize renderings of neural materials extracted by NeuMatEx (Fig. 5) and compare with PBR material extraction methods on synthetic and real images. We also investigate the impact of design choices. Please refer to the appendix and attached visualizer for more results.

6.1 Comparison with PBR Material Extraction Methods

For feed-forward PBR material extraction methods, we choose Hunyuan3D-2.1 (Tencent, 2025) and TRELLIS.2 (Xiang et al., 2026), using their texture reconstruction mode with known geometry. Note that Hunyuan3D-2.1 and TRELLIS.2 are conditioned on a *single* input view while our method uses multiple views; this comparison therefore highlights PBR's limitations rather than being a direct comparison. For a multi-view feed-forward method, we use LSRM (Li et al., 2026), which predicts PBR material and geometry, and further refine its PBR prediction with test-time optimization, denoted

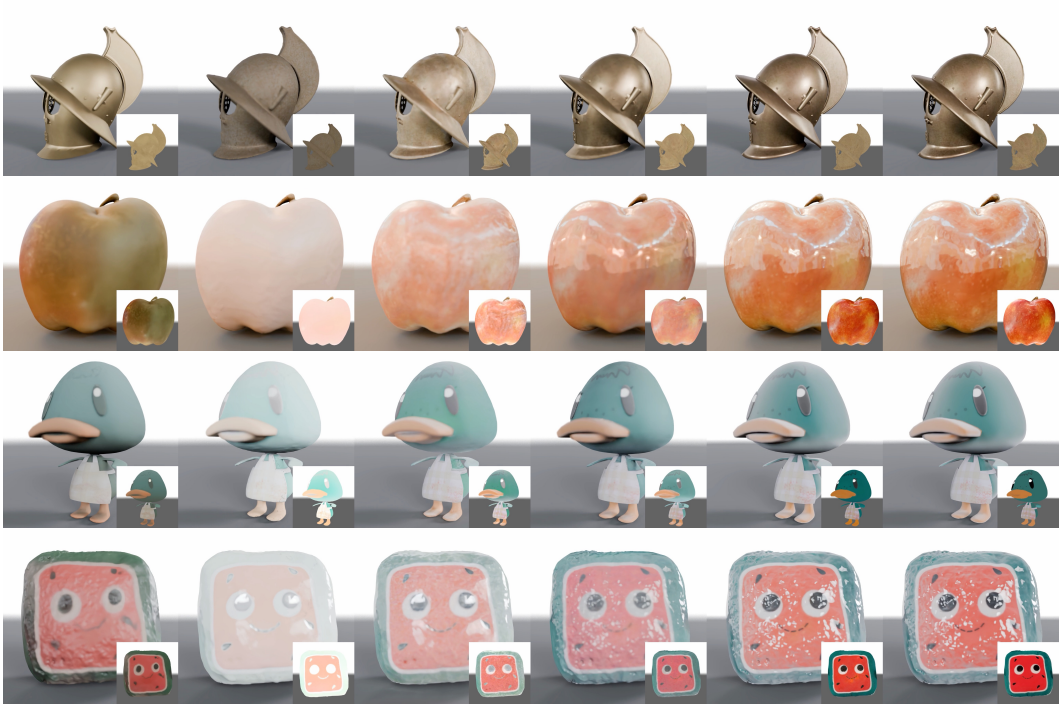

Figure 6: **Qualitative: Material Extraction.** All methods use known fixed geometry except LSRM (Li et al., 2026) and LSRM++, which predict their own; optimization methods use known poses and lighting. TRELLIS.2 (Xiang et al., 2026) is a monocular feed-forward PBR model, included to highlight PBR limitations rather than for a fair comparison. PBR-based methods fail to represent complex SVBSDFs, baking specular effects into the base color (see insets), while NeuMatEx decomposes them faithfully in the neural material latent space, yielding a clean base color.

Table 1: **Quantitative: Material Extraction.** For 40 test meshes with GT neural materials from Yu et al. (2026), we compare reconstruction metrics for renderings (RGB) and base color (BC), averaged over 120 orbit views. Relighting metrics are averaged over four held-out HDR probes.

	Config.					Recon. RGB			Recon. Base color		Relighting		
Method	Material	View	GT Mesh	Feed-fwd	TTO	PSNR↑	LPIPS↓	SSIM↑	siPSNR↑	PSNR↑	PSNR↑	LPIPS↓	SSIM↑
HUNYUAN3D-2.1	PBR	Single	✓	✓	✗	24.42	0.137	0.900	24.41	23.01	22.50	0.138	0.902
TRELLIS.2		Single	✓	✓	✗	23.55	0.140	0.912	24.60	23.95	23.91	0.138	0.914
LSRM		Multi	✗	✓	✗	18.29	0.190	0.870	22.15	12.54	18.33	0.190	0.873
LSRM++		Multi	✗	✓	✓	20.68	0.151	0.891	23.21	16.41	20.58	0.155	0.889
NVDIFFRECMC++		Multi	✓	✓	✓	26.25	0.097	0.938	26.31	24.89	27.50	0.097	0.937
NEUMATEX (Ours, LSRM geom.)	NM	Multi	✗	✓	✓	24.48	0.093	0.921	27.09	26.42	26.23	0.092	0.923
NEUMATEX (Ours, Full)		Multi	✓	✓	✓	34.78	0.050	0.963	31.45	25.30	33.13	0.047	0.966

LSRM++. We also extend Hasselgren et al. (2022) by initializing it with a PBR prediction from our PBR LMRM and optimizing only the materials, given known geometry and lighting. We denote this as NVDiffRecMC++. Both LSRM++ and NVDiffRecMC++ combine multi-view feed-forward PBR initialization with test-time optimization. Finally, NeuMatEx with LSRM geometry examines whether our method remains effective once the known geometry assumption is relaxed. We do not consider VideoNeuMat (Xue et al., 2026): it targets planar 2D materials and no code is available.

In Fig. 6, we show renderings of extracted materials alongside the diffuse base color for assets with complex specular effects. We use ground-truth neural materials in the basis of Yu et al. (2026) as a reference; all methods are rendered from the input view of TRELLIS.2 for a fair comparison. NeuMatEx reconstructs materials close to the reference, capturing the specular effects and separating the diffuse and specular terms without baking specular shading into the base color. In Tab. 1, we report reconstruction scores over 120 orbit views for both the path-traced renderings and base color, as well as relighting metrics under four held-out HDR probes. For base color, we additionally report scale-invariant PSNR (siPSNR), which factors out the inherently ambiguous global intensity scale of

Table 2: **Effect of the NeuMatEx design choices.** We ablate material parameterization (Param.), LMRM initialization (LMRM init.), and uncertainty regularization (Unc. reg.) on 40 held-out test meshes from Yu et al. (2026). Our full method (f) achieves the best material decomposition and competitive render quality, leveraging a large-scale trained prior and uncertainty-guided regularization.

	NeuMatEx Config.			Material Decomp. (↑)		Render Quality (↑)
	Param.	LMRM init.	Unc. reg.	$\text{PSNR}_{\text{BaseColor}}$	$\text{PSNR}_{\text{Latents}}$	$\text{PSNR}_{\text{PathTrace}}$
(a)		✗	✗	12.19±2.70	14.74±1.31	31.67±1.79
(b)	UV	✓	✗	22.73±3.71	25.40±3.67	35.67±1.80
(c)		✓	✓	23.67±3.16	26.32±4.17	33.94±2.57
(d)		✗	✗	13.56±3.50	17.15±2.56	28.67±3.03
(e)	Triplane	✓	✗	24.84±3.97	26.48±4.42	35.51±1.93
(f)		✓	✓	25.30±3.93	26.81±4.45	34.78±2.22

Table 3: **Ablation on TTO regularization.** We compare TTO results with and without uncertainty-guided regularization, evaluated on 40 held-out test meshes from Yu et al. (2026). Our predicted uncertainty (c) closely approaches the oracle upper bound (d), confirming the predicted uncertainty effectively guides TTO. Please refer to visual ablation results in Fig. S2.

	Material Decomp. (↑)		Render Quality (↑)
TTO Regularization Config.	$\text{PSNR}_{\text{BaseColor}}$	$\text{PSNR}_{\text{Latents}}$	$\text{PSNR}_{\text{PathTrace}}$
(a) No TTO (Only LMRM pred.)	23.82±3.72	26.21±4.03	31.99±2.95
(b) TTO *w/o* regularization	24.84±3.97	26.48±4.42	35.51±1.93
(c) TTO *w/* $\mathbf{G}_{\text{unc}}^{\text{LMRM}}$ (Ours)	25.30±3.93	26.81±4.45	34.78±2.22
(d) TTO *w/* $\mathbf{G}_{\text{unc}}^{\text{oracle}}$ (Ideal upper bound)	25.99±4.39	27.31±4.44	35.05±2.17

the recovered base color. NeuMatEx outperforms all PBR-based methods on both, showing that the expressiveness of neural materials is essential for appearance effects PBR cannot capture. Adding the same test-time optimization to the PBR baselines (LSRM++, NVDiffRecMC++) narrows but does not close this gap, indicating the limitation lies in the representation rather than the optimization. Finally, even with the imperfect predicted mesh, NeuMatEx extracts materials faithful enough to outperform the competing methods (LSRM, LSRM++) under the same predicted geometry by a large margin, showing that NeuMatEx generalizes beyond the known geometry setting.

6.2 Ablations

Impact of design choices. In Tab. 2, we ablate the design choices of NeuMatEx: the triplane material representation, the impact of the initial material prediction from the LMRM, and the uncertainty regularization. Neural materials are significantly harder to extract from multi-view images through inverse rendering, and we note that the strong prior from the LMRM initialization greatly improves both the render quality and the material decomposition compared to optimizing from randomly initialized materials ((a),(d) vs. (b),(e)). Uncertainty-guided optimization has a more subtle impact but increases the quality of the material decomposition ((b),(e) vs. (c),(f)). The triplane representation provides a significant quality improvement, since it enforces stricter spatial coherence than naïve UV texture mapping, which helps regularize test-time optimization ((a)-(c) vs. (d)-(f)).

Impact of uncertainty-guided regularization. In Tab. 3, we ablate the benefit of uncertainty-guided regularization for test-time optimization. The unconstrained TTO (row (b)) already outperforms the LMRM prediction alone, and adding uncertainty-guided regularization further improves material decomposition while retaining competitive render quality (row (c)). Also, we evaluate two regularization strategies using our predicted uncertainty $\mathbf{G}_{\text{unc}}^{\text{LMRM}}$, and an oracle $\mathbf{G}_{\text{unc}}^{\text{oracle}}$ that uses material G-buffer errors as uncertainty, *i.e.*, an ideal upper bound, up to scale (row (d)). Our TTO with predicted $\mathbf{G}_{\text{unc}}^{\text{LMRM}}$ performs close to the ideal case, supporting that LMRM produces well-calibrated uncertainty estimates that effectively guide TTO. In Fig. S2, we qualitatively show a clear improvement in material decomposition from uncertainty-guided regularization.

6.3 Neural Materials from Real-world Captures

Real-world objects exhibit complex reflection properties that go beyond the expressiveness of the standard PBR model. Extracting neural materials from real photos is a step toward data-driven SVBSDF authoring. As a proof-of-concept, we apply NeuMatEx to extract neural materials from



(a) Neural Material reconstruction on real-world captures **(b) Relighting real-world Neural Materials**

Figure 7: **Neural materials from real-world captures.** (a) We apply NeuMatEx to extract neural materials from real-world multi-view object capture dataset (Dong et al., 2025). Our extracted materials capture effects beyond the standard PBR, *e.g.*, clearcoat on the teapot. (b) Relit renderings of our extracted neural materials show that the recovered reflectance generalizes across illumination conditions, faithfully reproducing complex specular behavior without baked-in effects.



Figure 8: **Limitations.** We build on a pre-trained neural material basis (Yu et al., 2026), inheriting its limitations, where if predicted materials lie outside the valid domain of the basis, specular artifacts can appear. Our test-time optimization mostly mitigates this, but artifacts may still be observed.

real-world captures. We use the Digital Twin Catalog (DTC) (Dong et al., 2025) consisting of high-quality multi-view captures, meshes, camera poses, and environment lighting. We compare NeuMatEx against NVDiffRecMC++ in Fig. 7(a). The PBR materials struggle with complex glossy effects and layered reflectance, while our extracted neural materials are expressive enough to capture them, producing more faithful reflections that remain stable under novel illumination (Fig. 7(b)).

7 Discussion and Limitations

We present NeuMatEx, demonstrating that inverse rendering can recover expressive neural materials from images. Combining a strong feed-forward initialization from the Large Material Reconstruction Model with uncertainty-guided regularization stabilizes ill-posed test-time optimization in the neural material latent space. NeuMatEx outperforms PBR-based methods and recovers richer materials in synthetic and real-world scenarios, cleanly separating lighting from material response. While we build on the SVBSDF representation of Yu et al. (2026), our method can be extended to other neural material models. In future work we aim to improve LMRM material prediction quality and the resolution of the recovered material to capture finer appearance details.

Limitations. The complexity of the neural basis remains challenging even with careful initialization. In Fig. 8, LMRM predictions on out-of-domain inputs such as real-world photographs can exhibit specular artifacts with a red tint, as also observed by Yu et al. (2026). Test-time optimization often reduces this, but slight artifacts can remain. Learning a more robust neural material model, ideally from real-world observations, is an important direction toward photorealistic reflectance recovery.

AI USE STATEMENT

In this work, we used generative AI tools for the following tasks that require disclosure:

Support qualitative and thematic data analysis Generative AI tools assisted us in building the visualization framework used to inspect and compare results qualitatively, and in writing the scripts that compute our error metrics. The resulting comparisons and metrics were checked by the authors, and all interpretation of the outcomes is our own.

Implement the proposed method AI coding agents (Codex and Cursor) assisted with parts of the training and evaluation scripts and with refactoring. All generated code was reviewed, verified, and tested by the authors; the core method design is our own.

We have not used generative AI tools to generate synthetic datasets (our training data is produced entirely by a procedural, renderer-based pipeline), nor to clean or reformat datasets, to develop theoretical models or conceptual frameworks, to propose or refine hypotheses, to design or give feedback on our research methodology or experiments, to assist with translation, or to interpret results. The remaining tasks subject to required disclosure: formulating mathematical claims, supplying critical ingredients for proofs, and writing proofs, are not applicable to this work, as it contains no formal theorems or proofs.

We used generative AI tools for the following tasks for which disclosure is recommended:

Create or modify scientific figures or images We used generative AI tools to automate the assembly of collage-style result figures, which the authors then adjusted by hand. We note that every illustration in the paper was manually drawn by the authors, and every rendered result shown in the paper was produced by our method.

Create or edit software code AI coding agents (Codex and Cursor) were also used to build the supplementary `html` viewer.

Edit a research paper to improve readability We used an LLM to review a draft of this paper, and used the resulting feedback to manually correct grammatical errors and improve clarity.

We have reviewed all AI-assisted work. LLM-generated code was read, verified, and tested for correctness by the authors, and all AI-assisted text was rewritten or explicitly approved by the authors. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work does not involve human subjects, personally identifiable data, or the release of new datasets. All experiments are conducted on synthetic assets rendered in-house and on publicly available benchmarks, used in accordance with their licenses.

REPRODUCIBILITY STATEMENT

The two-stage pipeline of NeuMatEx is described in full in Secs. 3, 4 & 5. Architecture and training details of LMRM, including the backbone configuration, the triplane parameterization, and the heteroscedastic uncertainty formulation, are given in Sec. B.1, and the objective, hyperparameters, and optimization schedule of the test-time optimization stage are reported in Sec. B.2. The evaluation protocols, including baseline configurations and the details of all reported error metrics, are documented in Sec. 6. We also provide details about neural material rendering performance in Sec. B.4, and implementation details for real-world DTC benchmark are reported in Sec. B.5.

REFERENCES

Dejan Azinovic, Tzu-Mao Li, Anton Kaplanyan, and Matthias Nießner. Inverse path tracing for joint material and lighting estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

Sai Bi, Zexiang Xu, Pratul Srinivasan, Ben Mildenhall, Kalyan Sunkavalli, Miloš Hašan, Yannick Hold-Geoffroy, David Kriegman, and Ravi Ramamoorthi. Neural reflectance fields for appearance acquisition. *arXiv preprint, 2008.03824*, 2020. 2

Benedikt Bitterli, Petrik Clarberg, Chris Cummings, Aaron Lefohn, Steve Marschner, Jan Novák, Fabrice Rousselle, Andrea Weidlich, and Tizian Zeltner. Taming optimization variance in compact neural shading networks. In *ACM SIGGRAPH Conference Proceedings*, 2026. 3

Mark Boss, Varun Jampani, Kihwan Kim, Hendrik Lensch, and Jan Kautz. Two-shot spatially-varying BRDF and shape estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

Mark Boss, Varun Jampani, Raphael Braun, Ce Liu, Jonathan Barron, and Hendrik Lensch. Neural-pil: Neural pre-integrated lighting for reflectance decomposition. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, 2021. 2, 3

Brent Burley. Physically-Based Shading at Disney. In *ACM SIGGRAPH 2012 Courses: Practical Physically Based Shading in Film and Game Production*, 2012. 3

Wenzheng Chen, Joey Litalien, Jun Gao, Zian Wang, Clement Fuji Tsang, Sameh Khamis, Or Litany, and Sanja Fidler. DIB-R++: Learning to predict lighting and material with a hybrid differentiable renderer. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 2

Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A Universe of Annotated 3D Objects. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023. 5

Zhao Dong, Ka Chen, Zhaoyang Lv, Hong-Xing Yu, Yunzhi Zhang, Cheng Zhang, Yufeng Zhu, Stephen Tian, Zhengqin Li, Geordie Moffatt, Sean Christofferson, James Fort, Xiaqing Pan, Mingfei Yan, Jiajun Wu, Carl Yuheng Ren, and Richard Newcombe. Digital Twin Catalog: A Large-Scale Photorealistic 3D Object Digital Twin Dataset. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025. 9, 18, 23, 24

Jiahui Fan, Beibei Wang, Miloš Hašan, Jian Yang, and Ling-Qi Yan. Neural Layered BRDFs. In *ACM SIGGRAPH Conference Proceedings*, 2022. 1, 3

Jiahui Fan, Fujun Luan, Jian Yang, Miloš Hašan, and Beibei Wang. RNG: Relightable Neural Gaussians. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 3

Duan Gao, Xiao Li, Yue Dong, Pieter Peers, Kun Xu, and Xin Tong. Deep Inverse Rendering for High-Resolution SVBRDF Estimation from an Arbitrary Number of Images. *ACM Transactions on Graphics*, 38(4), 2019. 2

Jian Gao, Chun Gu, Youtian Lin, Zhihao Li, Hao Zhu, Xun Cao, Li Zhang, and Yao Yao. Relightable 3D Gaussian: Realistic Point Cloud Relighting with BRDF Decomposition and Ray Tracing. In *European Conference on Computer Vision (ECCV)*, 2024. 2

Andrew Gardner, Chris Tchou, Tim Hawkins, and Paul Debevec. Linear Light Source Reflectometry. *ACM Transactions on Graphics*, 22(3):749–758, 2003. 2

Abhijeet Ghosh, Tongbo Chen, Pieter Peers, Cyrus A. Wilson, and Paul Debevec. Estimating Specular Roughness and Anisotropy from Second Order Spherical Gradient Illumination. *Computer Graphics Forum*, 28(4):1161–1170, 2009. 2

Michael D. Grossberg and Shree K. Nayar. Modeling the space of camera response functions. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 26(10):1272–1282, 2004. 19

D. Guarnera, G. C. Guarnera, A. Ghosh, C. Denk, and M. Glencross. BRDF Representation and Acquisition. In *Proceedings of the 37th Annual Conference of the European Association for Computer Graphics: State of the Art Reports*, pp. 625–650, 2016. 2

Yu Guo, Cameron Smith, Miloš Hašan, Kalyan Sunkavalli, and Shuang Zhao. MaterialGAN: Reflectance Capture Using a Generative SVBRDF Model. *ACM Transactions on Graphics*, 39(6), 2020. 2

Jon Hasselgren, Nikolai Hofmann, and Jacob Munkberg. Shape, Light, and Material Decomposition from Images using Monte Carlo Rendering and Denoising. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 2, 3, 5, 7

Jon Hasselgren, Zheng Zeng, Miloš Hašan, and Jacob Munkberg. VideoMatGen: PBR Materials through Joint Generative Modeling. In *IEEE Conference on Computer Vision and Pattern Recognition Findings (CVPRF)*, 2026. 2

Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2D Gaussian Splatting for Geometrically Accurate Radiance Fields. In *ACM SIGGRAPH Conference Proceedings*. Association for Computing Machinery, 2024. 18

Yingwenqi Jiang, Jiadong Tu, Yuan Liu, Xifeng Gao, Xiaoxiao Long, Wenping Wang, and Yuexin Ma. GaussianShader: 3D Gaussian Splatting with Shading Functions for Reflective Surfaces. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

Haian Jin, Isabella Liu, Peijia Xu, Xiaoshuai Zhang, Songfang Han, Sai Bi, Xiaowei Zhou, Zexiang Xu, and Hao Su. TensoIR: Tensorial Inverse Rendering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

James T. Kajiya. The rendering equation. In *ACM SIGGRAPH Conference Proceedings*. ACM Press, 1986. doi: 10.1145/15922.15902. 17

Brian Karis. Real shading in Unreal Engine 4. *ACM SIGGRAPH Course on Physically Based Shading Theory and Practice*, 2013. 3

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 6

Peter Kocsis, Vincent Sitzmann, and Matthias Nießner. Intrinsic image diffusion for indoor single-view material estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

Peter Kocsis, Lukas Höllein, and Matthias Nießner. Intrinsic image fusion for multi-view 3d material reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. 2

Alexandr Kuznetsov, Krishna Mullia, Zexiang Xu, Miloš Hašan, and Ravi Ramamoorthi. NeuMIP: Multi-Resolution Neural Materials. *ACM Transactions on Graphics*, 40(4), July 2021. 1, 3

Hendrik P.A. Lensch, Jochen Lang, Asla M. Sa, and Hans-Peter Seidel. Planned sampling of spatially varying BRDFs. *Computer Graphics Forum*, 2003. 2

Zhengqin Li, Zexiang Xu, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Learning to reconstruct shape and spatially-varying reflectance from a single image. In *ACM SIGGRAPH Asia Conference Proceedings*, 2018. 2

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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

Zhengqin Li, Dilin Wang, Ka Chen, Zhaoyang Lv, Thu Nguyen-Phuoc, Milim Lee, Jia-Bin Huang, Lei Xiao, Yufeng Zhu, Carl S. Marshall, Yuheng Ren, Richard Newcombe, and Zhao Dong. LIRM: Large Inverse Rendering Model for Progressive Reconstruction of Shape, Materials and View-dependent Radiance Fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2

Zhengqin Li, Cheng Zhang, Jakob Engel, and Zhao Dong. LSRM: High-fidelity object-centric reconstruction via scaled context windows. In *European Conference on Computer Vision (ECCV)*, 2026. 2, 6, 7

Zhihao Liang, Qi Zhang, Ying Feng, Ying Shan, and Kui Jia. GS-IR: 3D gaussian splatting for inverse rendering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

Chih-Hao Lin, Jia-Bin Huang, Zhengqin Li, Zhao Dong, Christian Richardt, Tuotuo Li, Michael Zollhöfer, Johannes Kopf, Shenlong Wang, and Changil Kim. IRIS: inverse rendering of indoor scenes from low dynamic range images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2, 6, 19

Yehonathan Litman, Or Patashnik, Kangle Deng, Aviral Agrawal, Rushikesh Zawar, Fernando De la Torre, and Shubham Tulsiani. MaterialFusion: Enhancing Inverse Rendering with Material Diffusion Priors. In *International Conference on 3D Vision (3DV)*, 2025. 2, 5

Fujun Luan, Shuang Zhao, Kavita Bala, and Zhao Dong. Unified shape and svbrdf recovery using differentiable monte carlo rendering. *Computer Graphics Forum (Proceedings of EGSR)*, 40(4), 2021. 2

MaterialX Project. MaterialX: An open standard for network-based cg object looks. https://materialx.org/, 2026. Accessed: 2026-08-12. 1

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *European Conference on Computer Vision (ECCV)*, 2020. 17

Krishna Mullia, Fujun Luan, Xin Sun, and Miloš Hašan. RNA: Relightable Neural Assets. *ACM Transactions on Graphics*, 44(1), 2024. 3

Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Müller, and Sanja Fidler. Extracting Triangular 3D Models, Materials, and Lighting From Images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2, 3

Jacob Munkberg, Zian Wang, Ruofan Liang, Tianchang Shen, and Jon Hasselgren. VideoMat: Extracting PBR Materials from Video Diffusion Models. In *Eurographics Symposium on Rendering (EGSR)*, 2025. 2

Merlin Nimier-David, Zhao Dong, Wenzel Jakob, and Anton Kaplanyan. Material and Lighting Reconstruction for Complex Indoor Scenes with Texture-space Differentiable Rendering. In Adrien Bousseau and Morgan McGuire (eds.), *Eurographics Symposium on Rendering - DL-only Track*. The Eurographics Association, 2021. doi: 10.2312/sr.20211292. 2

William Peebles and Saining Xie. Scalable Diffusion Models with Transformers. In *IEEE International Conference on Computer Vision (ICCV)*, 2023. 4, 17

Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically Based Rendering: From Theory to Implementation*. MIT Press, 4th edition, 2023. 18

Nithin Raghavan, Krishna Mullia, Alexander Trevithick, Fujun Luan, Miloš Hašan, and Ravi Ramamoorthi. Generative Neural Materials. In *ACM SIGGRAPH Conference Proceedings*, 2025. 3

Gilles Rainer, Adrien Bousseau, Tobias Ritschel, and George Drettakis. Neural Precomputed Radiance Transfer. *Computer Graphics Forum*, 41(2), April 2022. 3

Carolin Schmitt, Simon Donne, Gernot Riegler, Vladlen Koltun, and Andreas Geiger. On joint estimation of pose, geometry and svBRDF from a handheld scanner. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

Maximilian Seitzer, Arash Tavakoli, Dimitrije Antic, and Georg Martius. On the Pitfalls of Heteroscedastic Uncertainty Estimation with Probabilistic Neural Networks. In *International Conference on Learning Representations (ICLR)*, 2022. 5

Yawar Siddiqui, Tom Monnier, Filippos Kokkinos, Mahendra Kariya, Yanir Kleiman, Emilien Garreau, Oran Gafni, Natalia Neverova, Andrea Vedaldi, Roman Shapovalov, and David Novotny. Meta 3D AssetGen: Text-to-Mesh Generation with High-Quality Geometry, Texture, and PBR Materials. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 2

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568(C), February 2024. 17

Team Wan. Wan: Open and advanced large-scale video generative models. *arXiv preprint, 2503.20314*, 2025. 4, 17

Tencent. Hunyuan3D 2.1: From Images to High-Fidelity 3D Assets with Production-Ready PBR Material. *arXiv preprint, 2506.15442*, 2025. 2, 6

Eric Veach and Leonidas J. Guibas. Optimally Combining Sampling Techniques for Monte Carlo Rendering. In *ACM SIGGRAPH Conference Proceedings*, 1995. 18

Zhengyi Wang, Yikai Wang, Yifei Chen, Chendong Xiang, Shuo Chen, Dajiang Yu, Chongxuan Li, Hang Su, and Jun Zhu. CRM: Single Image to 3D Textured Mesh with Convolutional Reconstruction Model. In *European Conference on Computer Vision (ECCV)*, 2024. 4, 16

Xinyue Wei, Kai Zhang, Sai Bi, Hao Tan, Fujun Luan, Valentin Deschaintre, Kalyan Sunkavalli, Hao Su, and Zexiang Xu. MeshLRM: Large Reconstruction Model for High-Quality Meshes. *arXiv preprint, 2404.12385*, 2024. 4

Michael Weinmann and Reinhard Klein. Advances in Geometry and Reflectance Acquisition (Course Notes). In *SIGGRAPH Asia 2015 Courses*, 2015. 2

Jianfeng Xiang, Xiaoxue Chen, Sicheng Xu, Ruicheng Wang, Zelong Lv, Yu Deng, Hongyuan Zhu, Yue Dong, Hao Zhao, Nicholas Jing Yuan, and Jiaolong Yang. Native and Compact Structured Latents for 3D Generation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. 2, 6, 7

Yingyan Xu, Gaspard Zoss, Prashanth Chandran, Markus Gross, Derek Bradley, and Paulo Gotardo. ReNeRF: Relightable Neural Radiance Fields with Nearfield Lighting. In *IEEE International Conference on Computer Vision (ICCV)*, 2023. 3

Bowen Xue, Saeed Hadadan, Zheng Zeng, Fabrice Rousselle, Zahra Montazeri, and Miloš Hašan. VideoNeuMat: Neural Material Extraction from Generative Video Models. In *ACM SIGGRAPH Conference Proceedings*, July 2026. 3, 4, 7

Kim Youwang, Tae-Hyun Oh, and Gerard Pons-Moll. Paint-it: Text-to-Texture Synthesis via Deep Convolutional Texture Map Optimization and Physically-Based Rendering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

Yunchen (Blaire) Yu, Jacob Munkberg, Jon Hasselgren, Chris Cummings, Steve Marschner, and Andrea Weidlich. Toward Richer Material Generation via Procedural Data Enhancement. In *ACM SIGGRAPH Conference Proceedings*, 2026. 1, 2, 3, 5, 7, 8, 9

Tizian Zeltner, Sébastien Speierer, Iliyan Georgiev, and Wenzel Jakob. Monte Carlo Estimators for Differential Light Transport. *ACM Transactions on Graphics*, 40(4):78:1–78:16, 2021. 18

Tizian Zeltner, Fabrice Rousselle, Andrea Weidlich, Petrik Clarberg, Jan Novák, Benedikt Bitterli, Alex Evans, Tomáš Davidovič, Simon Kallweit, and Aaron Lefohn. Real-time Neural Appearance Models. *ACM Transactions on Graphics*, 43(3), June 2024. 1, 2, 3, 18

Chong Zeng, Guojun Chen, Yue Dong, Pieter Peers, Hongzhi Wu, and Xin Tong. Relighting Neural Radiance Fields with Shadow and Highlight Hints. In *ACM SIGGRAPH Conference Proceedings*, 2023. 3

Zheng Zeng, Valentin Deschaintre, Iliyan Georgiev, Yannick Hold-Geoffroy, Yiwei Hu, Fujun Luan, Ling-Qi Yan, and Miloš Hašan. RGB↔X: image decomposition and synthesis using material-and lighting-aware diffusion models. In *ACM SIGGRAPH Conference Proceedings*, 2024. 2

Tianyuan Zhang, Zhengfei Kuang, Haian Jin, Zexiang Xu, Sai Bi, Hao Tan, He Zhang, Yiwei Hu, Miloš Hašan, William T. Freeman, Kai Zhang, and Fujun Luan. RelitLRM: Generative Relightable Radiance for Large Reconstruction Models. In *International Conference on Learning Representations (ICLR)*, 2025a. 4

Xiuming Zhang, Pratul P Srinivasan, Boyang Deng, Paul Debevec, William T Freeman, and Jonathan T Barron. NeRFactor: Neural Factorization of Shape and Reflectance Under an Unknown Illumination. *ACM Transactions on Graphics*, 2021a. 2

Yibo Zhang, Li Zhang, Rui Ma, and Nan Cao. TexVerse: A Universe of 3D Objects with High-Resolution Textures. *arXiv preprint, 2508.10868*, 2025b. 5

Yuanqing Zhang, Jiaming Sun, Xingyi He, Huan Fu, Rongfei Jia, and Xiaowei Zhou. Modeling indirect illumination for inverse rendering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2, 3

Yuxuan Zhang, Wenzheng Chen, Huan Ling, Jun Gao, Yinan Zhang, Antonio Torralba, and Sanja Fidler. Image GANs meet Differentiable Rendering for Inverse Graphics and Interpretable 3D Neural Rendering. In *International Conference on Learning Representations (ICLR)*, 2021b. 2

EXTRACTING NEURAL MATERIALS FROM IMAGES — SUPPLEMENTARY MATERIAL —

Anonymous authors
Paper under double-blind review

CONTENTS

A Supplementary Video & `HTML` Viewer **16**
B Implementation Details **16**
B.1 Large Material Reconstruction Model 16
B.2 Test-time Optimization 17
B.3 Neural Material Path Tracer 17
B.4 Neural Material Rendering Performance 18
B.5 NeuMatEx for Real-world Photos 18
C More Results **19**
C.1 Qualitative Ablation on Uncertainty-guided Regularization 19
C.2 Relighting Neural Materials Extracted by NeuMatEx 20
C.3 More Comparisons with PBR Material Extraction Methods 22
C.4 More Real-world Neural Material Extraction Results 23

A SUPPLEMENTARY VIDEO & HTML VIEWER

We encourage readers to watch the attached video and explore the results viewer:

- `NeuMatEx_supp.mp4`: We provide an overview of NeuMatEx and additional qualitative results.
- `html_visualizer/index.html`: An interactive viewer for side-by-side comparisons across methods, objects, and relighting conditions.

Note that, due to file-size constraints (100MB), the visual quality and resolution of the supplementary materials were reduced from our full results.

B IMPLEMENTATION DETAILS

B.1 LARGE MATERIAL RECONSTRUCTION MODEL

Multi-view input encoding. We encode $F = 17$ views, evenly spaced in a 360-degree orbit around the object, alongside the six canonical views. The 17 views are encoded using the frozen Wan2.1 VAE encoder, and the canonical views are each encoded with the image VAE for additional quality (Wang et al., 2024). The input views (F orbital + six canonical) are encoded into a latent tensor, $\mathbf{z}^{\mathbf{I}} \in \mathbb{R}^{(1+(F-1)/4+6)\times 16\times H/8\times W/8}$. We use $H = W = 512$ for the input image resolution. We process all the orbital input views jointly using Wan2.1's video VAE encoder, allowing the subsequent LMRM backbone to aggregate cross-view information.

Backbone. The backbone architecture of LMRM is a diffusion transformer (DiT) (Peebles & Xie, 2023), initialized from the pre-trained text-to-video Wan2.1-1.3B (Team Wan, 2025), repurposed as a *single-step* model. The backbone performs a single deterministic forward pass at a fixed timestep t=0, rather than iterative denoising, so that the DiT acts as a feed-forward regressor that takes multi-view images and predicts triplane feature planes. As the reconstruction task is not text-conditioned, we bypass the text cross-attention by feeding a fixed null-text embedding to every layer. We keep the original Wan2.1-1.3B configuration: 30 transformer blocks with model width 1536, 12 attention heads, feed-forward inner dimension 8960, patch embedding (1×2×2), and 3D rotary position embeddings (Su et al., 2024).

Triplane decoding head. After the encoded input latent tensor, $\mathbf{z}^{\mathbf{I}}$, is processed by the DiT backbone, the output tokens are mapped to triplane feature planes through a lightweight linear head followed by a convolutional decoder initialized from the pre-trained Wan2.1 VAE decoder (kept trainable). We decode the latent frames and obtain three triplane features: $\mathbf{T}_{\mathrm{XY}}, \mathbf{T}_{\mathrm{YZ}}, \mathbf{T}_{\mathrm{XZ}}$ representing the XY, YZ, and XZ planes, respectively, with 12 channels each. We decode each feature plane independently to avoid the causal temporal coupling of the video VAE decoder.

Triplane MLPs. Given a 3D surface point, we bilinearly sample the three axis-aligned planes and concatenate the resulting C=12-channel features ($3C$=36 total). We further append a NeRF-style positional encoding (Mildenhall et al., 2020) (3 frequency bands) for the 3D surface position and decode the material parameters with a triplane MLP, *i.e.*, a 4-layer MLP of width 64 with intermediate ReLU activations, and a final sigmoid (only for the material head). For the PBR pre-training stage, the material MLP head predicts 5 channels = base color (3D) + roughness (1D) + metallicity (1D). For the neural material fine-tuning stage, the material MLP head predicts 9 channels = base color (3D) + neural specular latents (6D). A parallel uncertainty MLP head, implemented as a separate MLP of identical capacity, predicts a per-channel log-variance used in a β negative log-likelihood (β-NLL) objective (Eq. 6). The MLPs are evaluated only at surface points via a gather-scatter scheme.

Training. We train the LMRM for 200K steps on 32 NVIDIA A100 GPUs with a per-GPU batch size of 2 in `bfloat16`, using AdamW with a cosine learning rate scheduler. For learning rates, we use 5×10^{-5} for the backbone, and 5×10^{-4} for both the material and uncertainty triplane MLPs. The β-NLL loss is disabled for the first 30K steps and then linearly ramped over 10K steps. Losses are computed on eight randomly sampled supervision views per object.

B.2 Test-time Optimization

In this stage, we optimize only the LMRM-initialized triplanes (the triplane MLPs are kept frozen) using Adam with a learning rate of 5×10^{-3}. During TTO, we use differentiable Monte Carlo path tracing with 128 samples per pixel (spp). For the uncertainty-guided material regularization, we cache the initial material predictions and per-channel aleatoric uncertainties from the frozen MLPs once before optimization and keep them fixed throughout. We set $\lambda_{\mathrm{reg}} = 2.0$, as this value was selected from a sweep over $\{0.1, 0.25, 0.5, 1, 2\}$ to best balance rendered-image and material-decomposition quality. Test-time optimization takes 15 minutes per object (1000 steps at 512×512 resolution) on a single NVIDIA A100 GPU, excluding final rendering. We use the Blender AgX tonemapper for optimization and for all results reported in the paper, unless otherwise noted.

B.3 Neural Material Path Tracer

For our path tracer with neural materials, the outgoing radiance $L(\omega_o, \mathbf{p})$ at a surface point $\mathbf{p}$ in direction ω_o is defined by the rendering equation (Kajiya, 1986):

$$L(\omega_o, \mathbf{p}) = \int_{\Omega} L_i(\omega_i, \mathbf{p}) f(\mathbf{p}, \omega_i, \omega_o)(\omega_i \cdot \mathbf{n}) d\omega_i. \tag{a}$$

This is an integral of the product of the incident radiance $L_i(\omega_i, \mathbf{p})$ from direction ω_i and the neural material SVBSDF $f(\mathbf{p}, \omega_i, \omega_o)$ (Eq. 1). The integration domain is the hemisphere Ω around the surface normal, $\mathbf{n}$. We evaluate the rendering equation using Monte Carlo integration:

$$L(\omega_o, \mathbf{p}) \approx \frac{1}{N} \sum_{i=1}^{N} \frac{L_i(\omega_i, \mathbf{p}) f(\mathbf{p}, \omega_i, \omega_o)(\omega_i \cdot \mathbf{n})}{p(\omega_i)}, \tag{b}$$

with samples drawn from some distribution $p(\omega_i)$.

We use *multiple importance sampling* (Veach & Guibas, 1995) (MIS), a framework for combining multiple sampling techniques to reduce variance in Monte Carlo integration. We apply MIS with two sampling techniques: light importance sampling, $p_{\text{light}}(\omega)$, using sampling from a piecewise-constant 2D distribution (Pharr et al., 2023), and BSDF sampling, $p_{\text{bsdf}}(\omega)$. The weighting functions $w_k(x)$ for each technique are chosen using the *balance heuristic* (Veach & Guibas, 1995; Pharr et al., 2023).

Following Zeltner et al. (2024), our BSDF sampler uses a frozen sampling decoder to extract parameters for two independent GGX lobes for the neural part, then selects one component among the diffuse cosine lobe and the two GGX lobes according to the predicted mixture weights (Eq. 2) and samples it. Using the taxonomy of differentiable Monte Carlo estimators (Zeltner et al., 2021), our importance sampling is *detached*, *i.e.*, gradients are not back-propagated to scene parameters in the sampling step, only in the material evaluation.

B.4 Neural Material Rendering Performance

We evaluate rendering performance of the neural materials at 1080p resolution on an RTX 5090 using the custom test scene in Fig. S1, which consists of four different neural materials and runs at 3.88 ms per frame at 1 spp in our path tracer with 10 bounces. Note that this number reflects the combined cost of both the neural evaluation and the underlying path-tracing computation. The universal neural material basis consists of MLPs with 4 layers of width 64 (4×64) for BSDF evaluation, a 3×64 auxiliary network for estimating transmission albedo, and a 2×32 sampling network for importance sampling. This clearly indicates that the neural material representation we extract allows for real-time performance within a path-tracing context. Please refer to Zeltner et al. (2024) for a thorough analysis of neural material runtime performance and implementation details.



Figure S1: **Rendering performance**. For a set of reference neural materials, the runtime performance for path tracing in 1080p resolution with 1 spp and 10 bounces is ∼4 ms on an RTX 5090.

B.5 NeuMatEx for Real-world Photos

We use the Digital Twin Catalog (DTC) (Dong et al., 2025) consisting of high-quality multi-view captures (∼ 120 photographs per object) together with meshes, camera poses, and measured environment lighting. First, we use novel-view synthesis to obtain the preset views, *i.e.*, 17 orbital views and 6 orthogonal views, needed for our LMRM. Specifically, we use 2D Gaussian Splatting (Huang et al., 2024), where we initialize the primitives at the mesh surface and the normals with the mesh normals. Then, we run our LMRM on the rendered views to estimate the initial triplane features for neural

materials. During the test-time optimization, we optimize against all real photograph views to avoid baking any novel-view-synthesis artifacts into the materials.

For all of these experiments, we use the captured mesh and HDR environment map, together with the posed LDR images. A constant tonemapping operator for the rendered images during our test-time optimization can introduce a domain gap with the real-world observations. To avoid this, we follow IRIS (Lin et al., 2025) and use a learnable camera response function (CRF) for these real-world experiments. We use the parametric CRF space of IRIS (Lin et al., 2025), based on Grossberg and Nayar's empirical model of response (EMoR) (Grossberg & Nayar, 2004), which fits a PCA basis to real-world measured CRFs. The optimized CRFs are then used to tonemap the rendered images for the comparisons as well as for the relightings.

C More Results

C.1 Qualitative Ablation on Uncertainty-guided Regularization

Figure S2 provides the qualitative ablation results for the proposed uncertainty-guided regularization in the test-time optimization. All configurations reproduce the path-traced renderings faithfully (first row), as the rendering objective alone can be satisfied by many material configurations. The difference instead appears in the decomposed material intrinsics. Our uncertainty-guided regularization anchors the optimization toward the LMRM prediction where it is confident, and relaxes it only where the predicted uncertainty is high. This keeps the base color and the specular latents free of baked-in lighting, while still recovering local details that the feed-forward prediction misses.

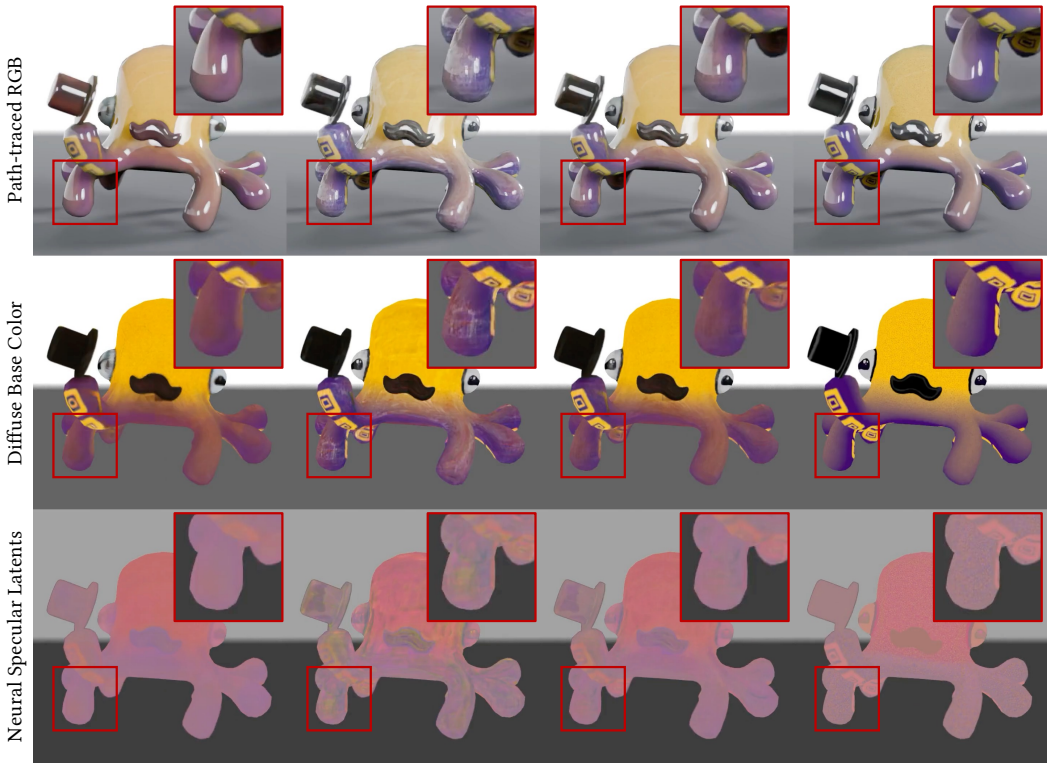

Figure S2: **Effects of uncertainty-guided regularization.** Unconstrained TTO (column 2) suffers from lighting baked into base color and neural latents. Our uncertainty-guided regularization (column 3) yields well-behaved and detailed material intrinsics over the LMRM initialization (see crops).

C.2 Relighting Neural Materials Extracted by NeuMatEx

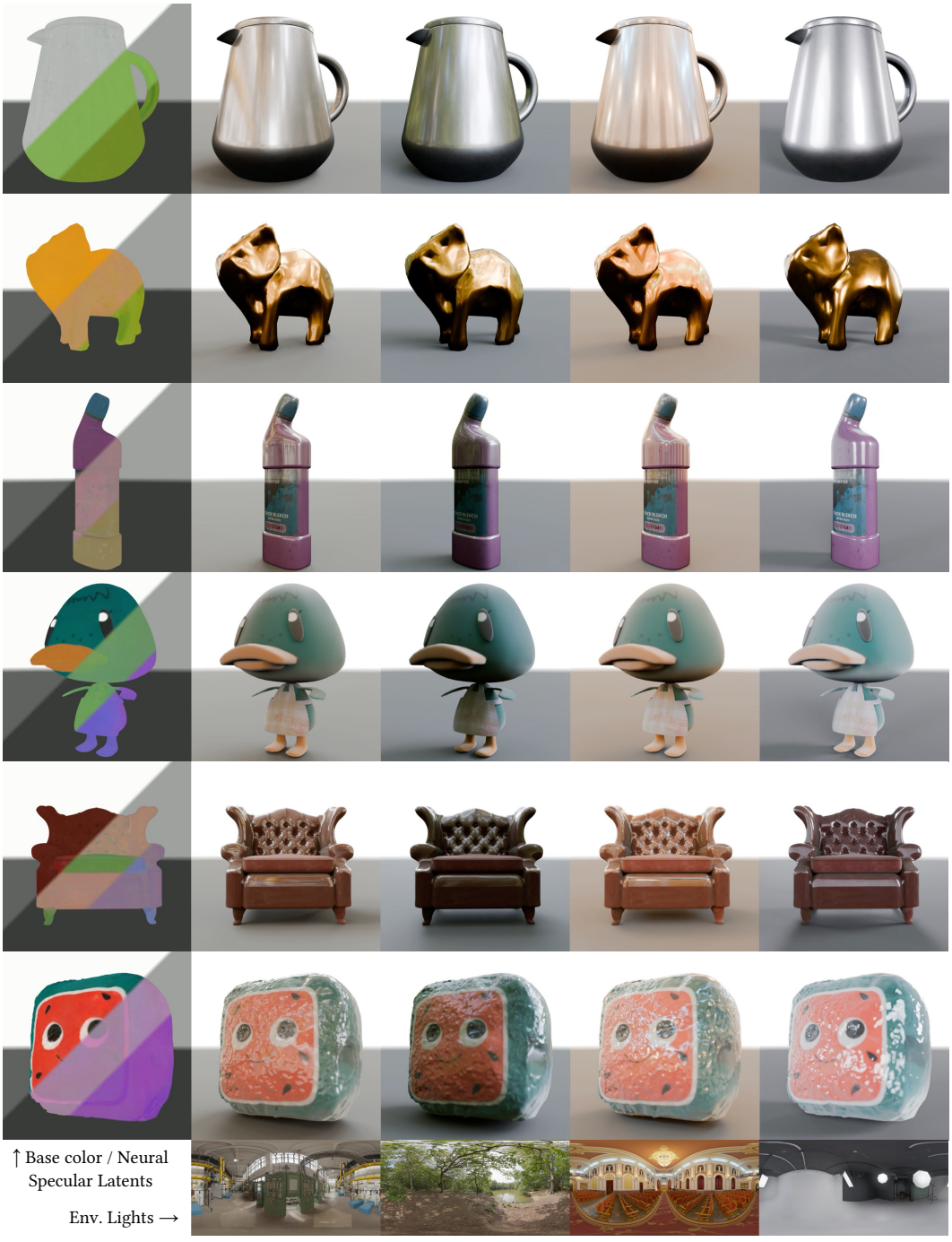

Figure S3: **Visualization of the extracted neural materials.** Column 1 shows the decomposed neural materials (diffuse base color + neural specular latents A & B), and columns 2-5 show the 128spp relit renderings using four different HDR light probes.



Figure S4: **Visualization of the extracted neural materials.** Column 1 shows the decomposed neural materials (diffuse base color + neural specular latents A & B), and columns 2-5 show the 128spp relit renderings using four different HDR light probes.

C.3 More Comparisons with PBR Material Extraction Methods

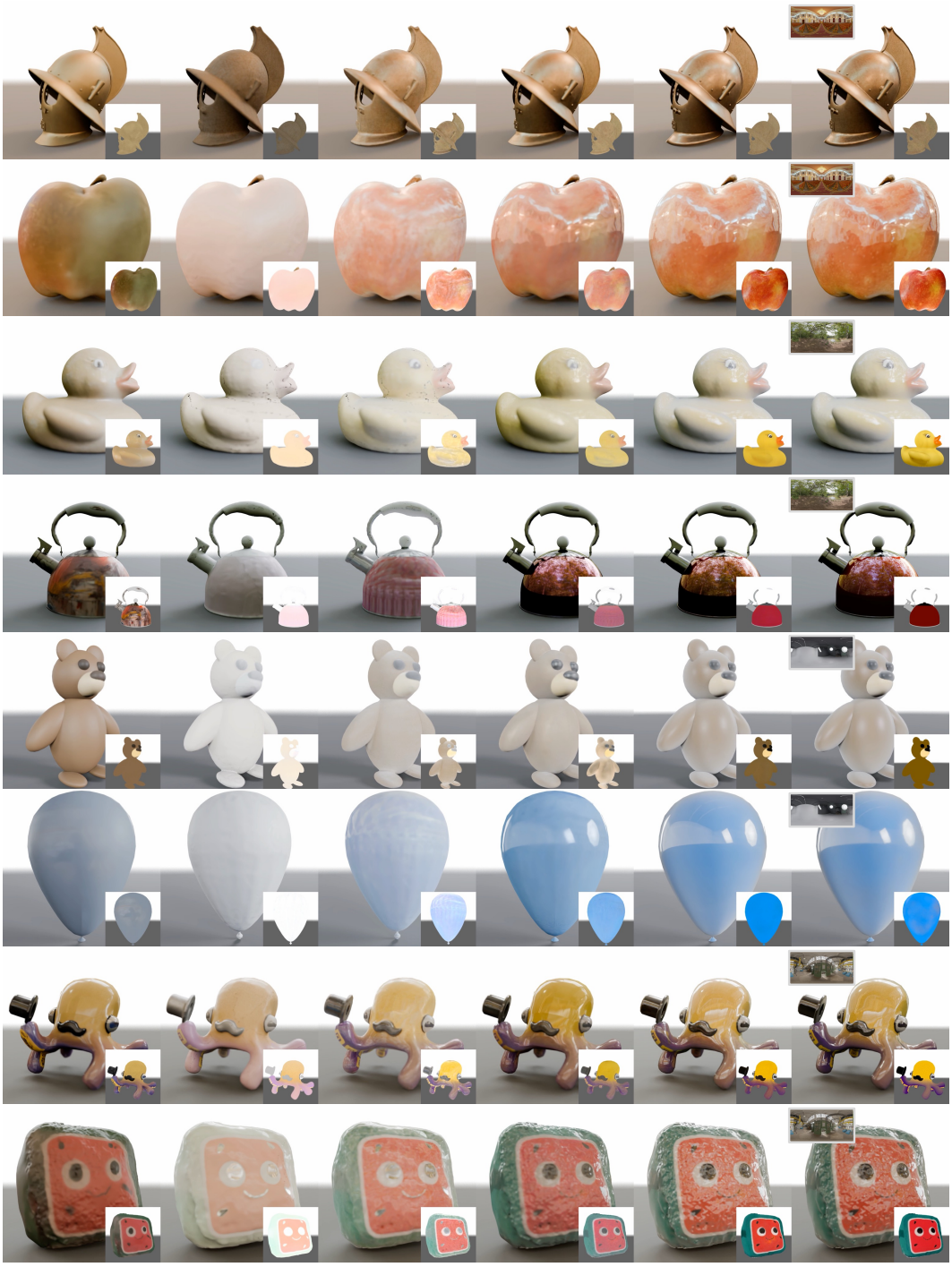

Figure S5: **PBR vs. Neural Material (128 spp, relit).** Insets: extracted diffuse base color (all columns, bottom), probes for relighting (last column, top).

C.4 More Real-world Neural Material Extraction Results



Figure S6: **Comparison for real-world neural material extraction.** Additional results on the DTC (Dong et al., 2025) dataset. Our method extracts more expressive neural materials than the PBR-based optimization method NVDiffRecMC++, capturing complex specular reflections.

We quantitatively compare the relighting performance of NVDiffRecMC++ and NeuMatEx, which indicates the quality of material decomposition. Note that the DTC benchmark is not designed with neural material reconstruction in mind: its ground-truth mesh assets are annotated with PBR materials, so the relit reference renderings cannot exceed the expressiveness of the PBR model. This setting inherently favors PBR-based methods, and we therefore expect only a small gap between the two approaches. Nevertheless, our extracted materials better capture specular effects and outperform NVDiffRecMC++ on all relighting metrics in Tab. S1.

Table S1: Relighting performance on the DTC benchmark. Results are averaged over four HDR environments and eight assets. Best results are shown in bold.

Method	Relighting		
	PSNR ↑	LPIPS ↓	SSIM ↑
NVDIFFRECMC++	23.70	0.039	0.951
NEUMATEX (Ours, Full)	**24.03**	**0.037**	**0.952**

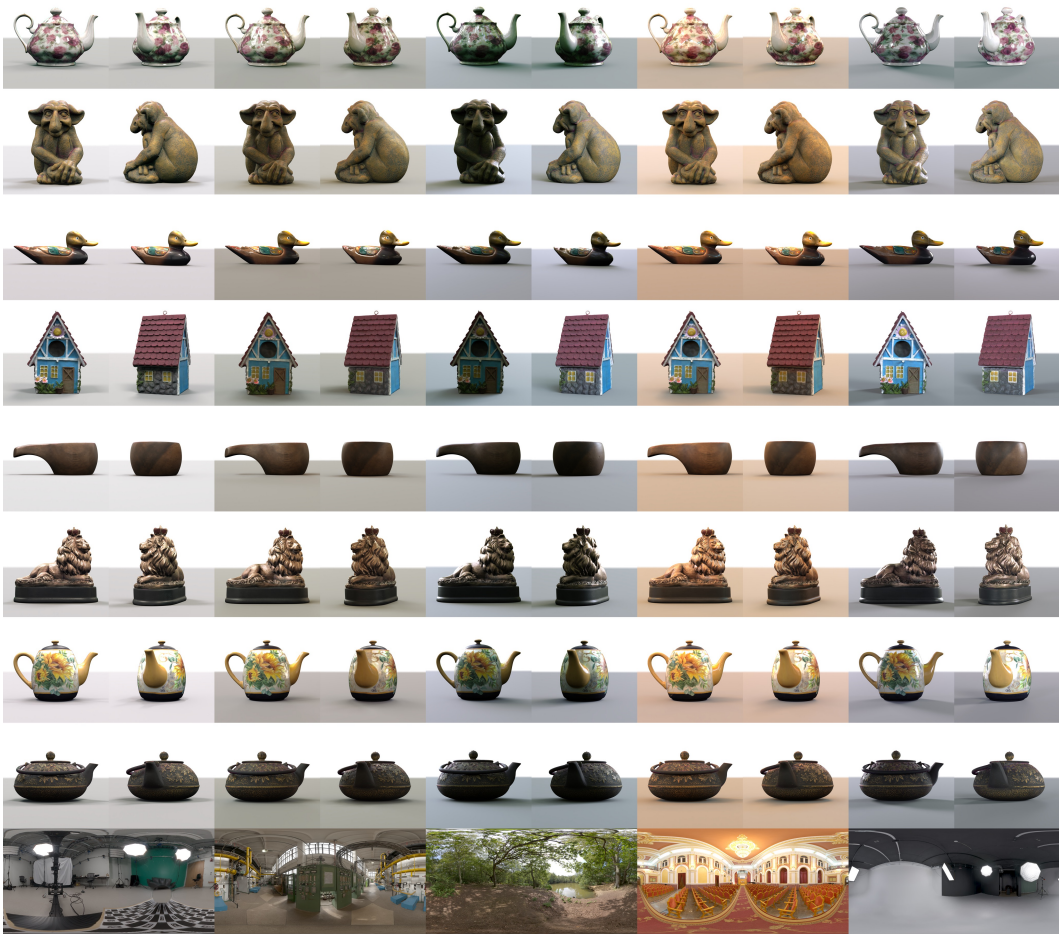

Figure S7: **Relighting real-world neural materials.** Relighting results showing two views of multiple objects from the DTC (Dong et al., 2025) dataset. Our extracted neural materials generalize to novel lighting conditions producing complex specular effects.