STYLE2SHAPE: IMAGE STYLE GUIDED 3D SHAPE MATERIAL GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

We present **Style2Shape**, a framework for generating editable physically-based rendering (PBR) materials for a 3D mesh from a single reference image. Compared with text-guided methods, images provide high-bandwidth appearance cues but introduce two key difficulties: (i) faithfully transferring fine-grained visual style while preserving the target shape's structure, and (ii) producing physically-plausible, standard PBR assets rather than implicit, non-editable representations. To address these challenges, we propose a hybrid material representation that couples a retrieved procedural material (for physically-consistent reflectance) with a generated texture (for rich, instance-specific details), combined via learnable blending. Our pipeline consists of three stages: (1) **structure-guided appearance transfer** to synthesize pixel-aligned supervision, (2) **hybrid initialization** via physics-based retrieval and texture generation, and (3) **physics-based optimization** that jointly refines procedural parameters, textures, and blending weights through differentiable rendering. Experiments show that Style2Shape produces high-fidelity, physically-plausible materials exported as standard PBR maps, making the results directly compatible with existing rendering workflows.

Keywords: 3D material generation, physically-based rendering, differentiable rendering, procedural materials, texture synthesis.

1 INTRODUCTION

Realistic material acquisition is central to computer graphics and vision, with broad applications in design, AR/VR, robotics, and digital content creation. Despite rapid progress in differentiable rendering and generative models, recovering high-quality physically-based rendering (PBR) materials for a target 3D mesh from minimal observations—in particular, a single reference image—remains challenging.

This problem is fundamentally ill-posed because geometry, illumination, and reflectance jointly explain image appearance. For example, high intensity may arise from either a bright albedo or strong specular lighting. In addition, the reference image and target mesh often differ in viewpoint and structure, making direct texture projection unreliable and motivating methods that can transfer appearance while respecting the target geometry.

Our objective differs from single-view reconstruction or novel-view synthesis. The target geometry is assumed to be available; the desired output is an explicit set of material maps that can be inspected, edited, tiled, and rendered under lighting conditions other than the one observed in the reference. Consequently, matching the reference image in a single view is necessary but not sufficient. A useful solution must also separate appearance into physically interpretable channels and maintain coherent behavior when the camera or illumination changes. This requirement motivates the use of physically-based image formation rather than treating material acquisition purely as an image stylization problem.

Existing approaches exhibit clear trade-offs. Text-guided methods Chen et al. (2023); Michel et al. (2022); Zhang et al. (2024) are flexible but often underspecified for materials, where small changes in wording can lead to large appearance changes and fine-grained patterns are hard to control. Image-guided methods Liu et al. (2023; 2024) provide more precise cues, yet commonly yield implicit or non-editable results that do not fit standard PBR workflows. Procedural materials Shi et al.

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.

(2020); Hu et al. (2022); Yan et al. (2023) offer editability and physically-meaningful parameters, but their expressivity is limited by the predefined parameter space of the material graph, which is often insufficient to reproduce the diversity of real-world textures.

These observations motivate a hybrid question: *can we combine the physical consistency and editability of procedural materials with the expressive power of generative textures, so as to obtain materials that are both plausible and visually faithful?*

We answer this question with **Style2Shape**, a hybrid material optimization framework. The key idea is to represent each material segment as a learnable blend of (i) a procedural component that anchors reflectance parameters (e.g., roughness and metallicity) and (ii) a generated texture that captures instance-specific visual detail. We then jointly optimize all components in a physics-based differentiable rendering loop, producing results that are both physically grounded and closely match the reference appearance.

Our framework consists of three stages:

(1) Structure-guided appearance transfer: we cast appearance transfer as constrained image-to-image translation to synthesize a geometrically-aligned supervision image that preserves the target structure while adopting the reference style.

(2) Hybrid PBR initialization: we retrieve a procedural material to ground physical reflectance and generate a complementary texture to capture fine-grained patterns.

(3) Physics-based optimization: we jointly refine procedural parameters, generated textures, and blending weights via differentiable rendering, ensuring the final appearance follows a physically-consistent image formation process.

The main contributions of our work are as follows:

- We propose a unified framework that generates *editable* PBR materials for a target 3D mesh from a single reference image, bridging generative priors and physically-based rendering.
- We introduce a hybrid material representation that blends procedural reflectance (physically-meaningful parameters) with generated textures (expressive appearance detail) via learnable weights.
- We develop a physics-based optimization scheme that jointly refines all components through differentiable rendering, preserving physical plausibility while matching the reference.

2 Related Work

2.1 Image- and Text-Guided Material Generation

Text-guided methods such as Text2Mesh Michel et al. (2022), Fantasia3D Chen et al. (2023), and Paint-it Youwang et al. (2024) generate or optimize appearance from language; MATLABER Xu et al. (2023) incorporates BRDF auto-encoders, while Material Anything Huang et al. (2025) predicts PBR maps with a diffusion model and a rendering loss. Text is flexible but can under-specify the precise patterns and reflectance of a particular reference. Image guidance supplies richer appearance evidence: Material Palette Lopes et al. (2024) estimates SVBRDF maps from an image, and Make-It-Real Fang et al. (2024) uses multimodal analysis and retrieval to assign PBR materials to 3D assets. Image-conditioned approaches such as Zero-1-to-3 Liu et al. (2023) and One-2-3-45 Liu et al. (2024) instead emphasize novel views or geometry, not reusable PBR assets. Direct image-to-map estimation must still resolve the mismatch between a reference photograph and the target mesh, while retrieval alone is bounded by its database. We use image editing to align reference appearance to target geometry, then recover explicit, editable PBR maps rather than treating the edited image as the final material.

2.2 Differentiable Rendering and Procedural Materials

Physics-based differentiable rendering enables material optimization against images. Edge sampling Li et al. (2018) and path reparameterization Loubet et al. (2019) address gradients at visibil-

ity discontinuities; Mitsuba 3 provides a practical implementation Jakob et al. (2022). Multi-view inverse rendering can jointly optimize geometry, materials, and lighting Munkberg et al. (2022), whereas we assume geometry is given and infer materials from a single style reference. Procedural graph fitting offers resolution independence and editable parameters: MATch/DiffMat Shi et al. (2020) makes material graphs differentiable, neural proxies extend optimization to non-differentiable nodes Hu et al. (2022), and PSDR-Room Yan et al. (2023) and Mapa Zhang et al. (2024) combine procedural materials with inverse rendering. These methods show the value of physical priors, but a fixed graph cannot represent every instance-specific print or grain. Our method retains a retrieved procedure as the physical base and adds segment-specific generated albedo detail for patterns absent from the library, optimizing both through differentiable rendering.

3 METHOD

3.1 PROBLEM SETTING AND OUTPUT REPRESENTATION

Given a UV-mapped target mesh S partitioned into K material segments $\{S_k\}_{k=1}^{K}$ and a reference image I_{ref} of the desired appearance, we output editable albedo, roughness, metallic, and normal maps for each segment. The reference may depict a different object or viewpoint. If material groups are unavailable, mesh segmentation such as SAMesh Tang et al. (2024) can be applied beforehand. The optimized variables are $\Theta = \{E, \theta_{\text{uv}}, \theta_{\text{proc}}, w\}$: environment lighting, UV transform, procedural parameters, and the procedural/generated albedo blend weight.

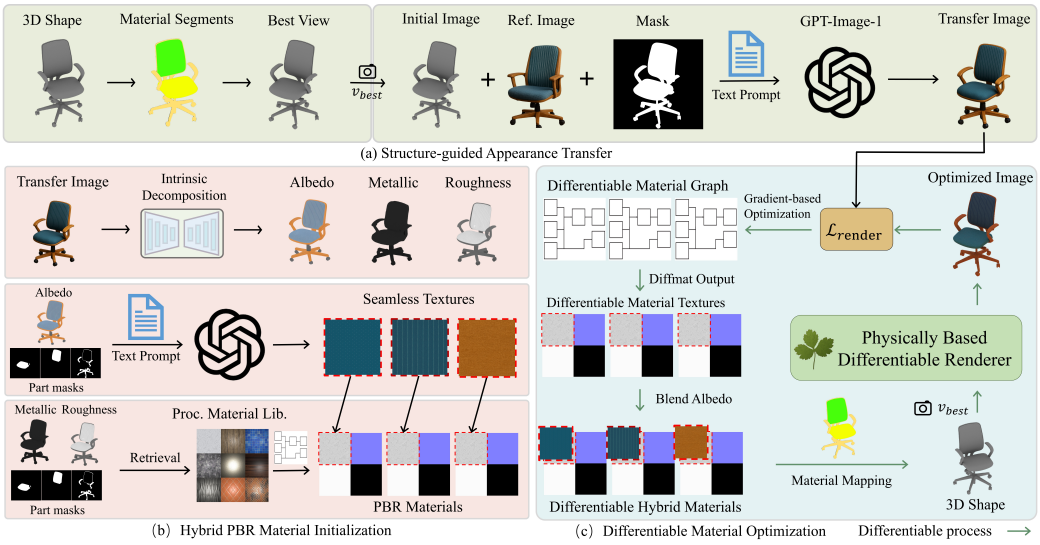

Figure 1: Our three-stage pipeline for generating PBR materials from a single image. (a) We synthesize a geometrically-aligned target image that transfers the reference style. (b) A hybrid material is initialized by blending a procedural base with a generated seamless texture. (c) The material is then optimized within a differentiable renderer to match the target.

3.2 STAGE 1: STRUCTURE-GUIDED APPEARANCE TRANSFER

3.2.1 MULTI-MODAL INPUT CONSTRUCTION.

We sample n viewpoints on a sphere and render a per-segment visibility mask for each. Let $a_k(v)$ be the visible screen-space area of segment S_k in view v. We retain the views with the largest number of visible segments as $V_{\text{candidate}}$, then select the one that maximizes the area of its least visible segment:

$$v_{\text{best}} = \arg \max_{v \in V_{\text{candidate}}} \min_{k: a_k(v) > 0} a_k(v). \tag{1}$$

From this viewpoint, we render the untextured target image I_{geom} for structure-guided transfer.

3.2.2 STRUCTURE-GUIDED IMAGE EDITING.

An image editing model (e.g., GPT-Image-1) receives I_{geom} as the structural target and I_{ref} as the appearance reference. Its prompt requires the target's silhouette, geometry, and part layout to remain fixed while transferring color, texture, and reflectance cues; it forbids introducing new structures. We project the edited appearance I_{edit} onto the target using its rendered mask at v_{best}, obtaining the pixel-aligned supervision image I_{tr}. This image is an optimization target, not the final texture; Appendix B provides the prompt wording.

3.3 STAGE 2: HYBRID PBR MATERIAL INITIALIZATION

The retrieved procedure supplies the PBR channels and anchors roughness, metallicity, and normal variation; a generated texture adds reference-specific albedo detail. We restrict generation to albedo to avoid directly trusting generated non-albedo maps whose errors could produce implausible reflectance.

3.3.1 INTRINSIC IMAGE DECOMPOSITION

Before material retrieval, RGB-X Zeng et al. (2024) decomposes I_{tr} into albedo, normal, roughness, and metallic maps, denoted I_{tr}^{alb}, I_{tr}^{nor}, I_{tr}^{rgh}, and I_{tr}^{met}. We denote this decomposition by D and use its estimates as retrieval cues.

3.3.2 PHYSICS-BASED MATERIAL RETRIEVAL

For each segment S_k and procedural candidate $P_m \in \mathcal{P}$, we UV-map and project its roughness and metallic textures from v_{best}, then compare them with the corresponding decomposed regions:

$$P_k^* = \arg\min_{P_m \in \mathcal{P}} \left[\lambda_{\text{rgh}} \| I_{tr,k}^{\text{rgh}} - P_k(\theta_{\text{rgh}}^m) \|_2^2 + \lambda_{\text{met}} \| I_{tr,k}^{\text{met}} - P_k(\theta_{\text{met}}^m) \|_2^2 \right] \tag{2}$$

Here $P_k(\cdot)$ projects a UV-mapped parameter texture into the screen-space region of S_k; $I_{tr,k}^{\text{rgh}}$ and $I_{tr,k}^{\text{met}}$ are the corresponding decomposed maps. We use roughness and metallicity rather than albedo because instance-specific patterns may be absent from the library. Small segments receive uniform material initialization without retrieval.

3.3.3 GENERATIVE TEXTURE SYNTHESIS AND BLENDING

For each segment, we provide its masked region in I_{tr} to the image editing model to generate a tileable albedo texture $T_{A,k}^{\text{gen}}$. Appendix B gives the texture prompt. We blend it with the retrieved procedural albedo $T_{A,k}^{\text{proc}}$:

$$T_{A,k}^{\text{final}} = (1 - w_k) \cdot T_{A,k}^{\text{proc}} + w_k \cdot T_{A,k}^{\text{gen}}, \tag{3}$$

where $w_k \in [0, 1]$ is a learnable blending weight for segment S_k.

3.4 STAGE 3: DIFFERENTIABLE MATERIAL OPTIMIZATION

We refine E, the UV transform $\theta_{\text{uv}} = \{s, r, t\}$ (scale, rotation, translation), and material parameters $\theta_{\text{mat}} = \{\theta_{\text{proc}}, w\}$ in a coarse-to-fine differentiable rendering loop. Separating these updates prevents illumination, texture placement, and albedo from compensating for one another before their initial estimates are stable.

3.4.1 PROGRESSIVE OPTIMIZATION STRATEGY

Step 1: Lighting Estimation. We optimize the environment map E against I_{tr} while keeping the initialized material and UV transform fixed, obtaining $\hat{E}$. This prevents the material albedo from absorbing illumination differences.

Step 2: UV Alignment. With $\hat{E}$ fixed and $w = 1$, we search discrete UV rotations and scales using VGG Gram-matrix distance Gatys et al. (2016), then continuously refine θ_{uv}. Setting $w = 1$ exposes

the generated pattern during alignment instead of letting the procedural component dominate its statistics.

Step 3: Joint Material Refinement. Keeping $\hat{E}$ fixed, we jointly refine θ_{mat} and θ_{uv} by minimizing the total objective below. The procedural parameters control reflectance and reusable structure, while the generated albedo and w account for reference-specific detail. We use 50/50/100 iterations for these three steps, respectively; learning rates and sampling settings appear in Section 4.2.

3.4.2 Optimization Objective

For $I_{\text{render}} = R(S, \theta_{\text{mat}}, \theta_{\text{uv}}, E)$, we combine a global reconstruction term with a style term for each segment:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{recon}} + \sum_{k=1}^{K} \lambda_k \mathcal{L}^k_{\text{style}}(I_{\text{render},k}, I_{tr,k}) \tag{4}$$

Here $\mathcal{L}_{\text{recon}} = \lambda_{\text{global}} \| D(I_{\text{render}}) - D(I_{tr}) \|_1$ compares decomposed appearance channels.

The segment-wise style term compares VGG-16 feature Gram matrices across selected layers $\mathcal{F}$. For each material segment, we use the weighted squared Frobenius distance

$$\mathcal{L}^k_{\text{style}}(I_{\text{render},k}, I_{tr,k}) = \sum_{l \in \mathcal{F}} \frac{\omega_l}{C_l^2} \| \mathcal{G}_l(I_{\text{render},k}) - \mathcal{G}_l(I_{tr,k}) \|_F^2, \tag{5}$$

where C_l is the feature-channel count, ω_l weights layer l, and $\mathcal{G}_l$ is its Gram matrix. Applying this loss per segment prevents a large region from explaining the texture statistics of smaller materials; the Gram-matrix definition is in Appendix B.

4 Experiments

4.1 Dataset and Material Library

Target meshes from 3DCoMPaT Li et al. (2022) and 3D Warehouse Trimble Inc. (2024) span bags, beds, buildings, chairs, sofas, and tables. Each category has at least five models and five CC0 reference photographs. Main experiments pair targets with semantically related but geometrically distinct references; cross-category transfers appear in Appendix A.

The procedural library contains 263 SBS materials authored in Adobe Substance 3D Designer Adobe Inc. (2024). Five variants per material vary roughness, scale, and color, yielding 1,315 retrieval candidates at 512×512 resolution. Every target segment searches the same library.

4.2 Implementation Details.

Our physics-based optimization pipeline is built upon the **Mitsuba 3** differentiable renderer Jakob et al. (2022). For propagating gradients through the rendering process, we utilize the **reparameterization** technique of Loubet et al. Loubet et al. (2019), which handles discontinuities from visibility events. We employ a **path tracing integrator** for all rendering operations. The rendering resolution is set to 512×512 pixels. We use an independent sampler with 64 samples-per-pixel (spp) and apply $8\times$ downsampling when computing the rendering loss to reduce Monte Carlo noise and optimization cost.

Using Adam Kingma & Ba (2014), we first run lighting estimation for 50 iterations with learning rate 10^{-2}. UV initialization then searches rotation angles $\{0, \pi/4, \pi/2, -\pi/2, -\pi/4\}$ and texture scales $\{0.5, 1, 2, 4, 8\}$, after which the UV transform is refined for 50 iterations at learning rate 10^{-3}. Joint material refinement runs for 100 iterations, with a cosine schedule decreasing the material learning rate from 10^{-2} to 10^{-4}. The full pipeline takes approximately 10 minutes per asset on one NVIDIA RTX 3090 GPU.

4.3 Qualitative Evaluation

We compare PSDR-Room Yan et al. (2023) and Material Palette Lopes et al. (2024) using the same Stage 1 transfer image supplied to our method. This shared input isolates differences in how

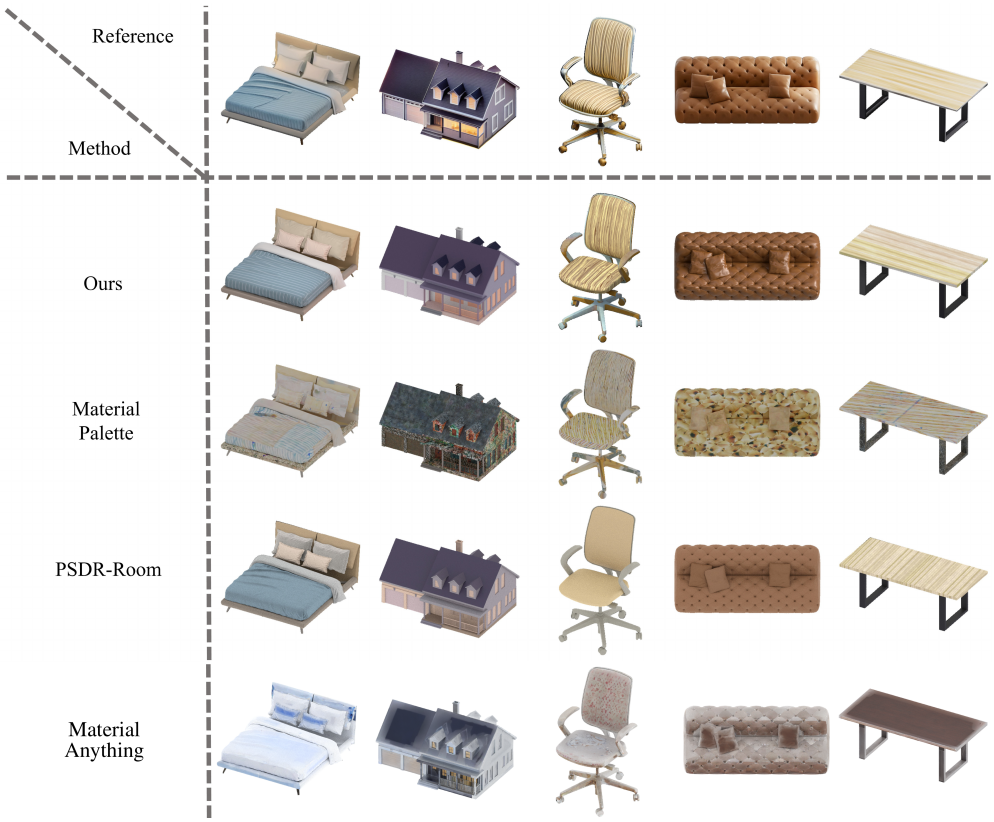

Figure 2: Qualitative comparison on the same five target meshes and fixed views (bed, building, chair, sofa, and table). PSDR-Room and Material Palette receive our structure-aligned Stage 1 transfer image; Material Anything receives a text prompt derived from the corresponding original reference photograph. Its quantitative results are included in Table 1.

each material representation converts aligned image evidence into a reusable asset. Material Anything Huang et al. (2025) instead receives text prompts derived from the reference photographs. All outputs are rendered on the same target meshes and evaluation views (Fig. 2).

Comparison with Baselines. Figure 2 highlights complementary limitations: PSDR-Room preserves coarse color and highlights but misses patterns absent from its procedural graph, while Material Palette captures local variation that can misalign with the target UV layout. Material Anything follows the geometry in some regions but differs from the reference in color and material assignment. Our hybrid results combine procedural reflectance with reference-specific fabric, wood, and decorative detail.

Material Generation Fidelity. Figure 3 shows materials across object categories, including leather and wood. Beds and sofas expose texture seams and scale errors on broad surfaces, while chairs, tables, and buildings challenge thin parts and multiple material segments. These examples test whether segment-level fitting remains effective across different mesh topologies rather than relying on a single asset layout. Appendix A shows additional styles on one geometry and cross-category transfers.

Multi-View Consistency. Figure 4 shows texture orientation and scale remaining attached to the surface under novel views, while specular appearance changes with view and lighting through the explicit BRDF. Unlike a view-specific painted result, the exported UV-space maps are reused without adjustment when the camera or illumination changes.



Figure 3: Rendered materials on target meshes from different object categories.



Figure 4: Renderings from novel viewpoints and under varied lighting conditions, showing the appearance of the recovered materials across views.

4.4 Quantitative Evaluation

PSNR, SSIM Wang et al. (2004), and LPIPS Zhang et al. (2018) compare renderings with Stage 1 supervision in terms of pixel, structural, and perceptual similarity, respectively.

Rendering Quality Assessment. Table 1 reports overall scores of 16.549 PSNR, 0.575 SSIM, and 0.160 LPIPS for our method, versus 16.199, 0.484, and 0.182 for PSDR-Room. Our method leads on beds, chairs, sofas, and tables; PSDR-Room has higher PSNR and SSIM on bags, where LPIPS is tied at 0.193. Text-conditioned Material Anything obtains 12.198 PSNR, 0.409 SSIM, and 0.162 LPIPS overall.

Table 1: Quantitative evaluation of rendering quality against structure-aligned transfer images. Higher PSNR/SSIM and lower LPIPS indicate better image fidelity. Best values are in **bold**.

Category	Ours			PSDR-Room			Material Palette			Material Anything (text)		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
All	**16.549**	**0.575**	**0.160**	16.199	0.484	0.182	13.071	0.456	0.203	12.198	0.409	0.162
Bags	15.499	0.455	**0.193**	**16.614**	**0.516**	**0.193**	13.562	0.469	0.223	14.420	0.438	0.221
Beds	**17.440**	**0.658**	0.172	16.854	0.460	0.216	14.845	0.554	0.204	15.719	0.511	**0.156**
Buildings	**14.239**	**0.395**	0.220	14.162	0.367	0.225	12.823	0.209	0.247	8.866	0.297	**0.198**
Chairs	**15.232**	**0.461**	**0.128**	15.187	0.413	0.153	13.459	0.376	0.150	11.632	0.359	0.150
Sofas	**15.723**	**0.640**	0.175	15.030	0.535	0.195	12.893	0.542	0.224	10.824	0.431	**0.151**
Tables	**18.666**	**0.623**	0.124	18.051	0.499	0.149	12.304	0.441	0.182	11.725	0.419	**0.096**

Table 2: User study results. We report the average user preference scores on a 5-point scale (1=Poor, 5=Excellent). Each row represents a material generation method.

Method	**Shape Category**						**Overall Avg.**
	Bags	Beds	Buildings	Chairs	Sofas	Tables	
Ours	**3.56**	**4.07**	**3.96**	**3.80**	**4.20**	**3.83**	**3.90**
PSDR-Room	3.37	3.32	3.53	3.40	3.17	3.44	3.37
Material Palette	2.77	1.95	1.78	2.38	2.53	2.20	2.27
Material Anything	3.00	3.00	2.78	2.89	2.67	2.67	2.83

User Study. For the user study, 30 participants blindly rated four methods on 54 cases (three models and three references for each of six categories). Each trial displayed the reference and randomized results; participants rated material-appearance similarity in color, texture, and reflectance on a 5-point scale. Our mean score is 3.90, versus 3.37 for PSDR-Room, 2.83 for Material Anything, and 2.27 for Material Palette (Table 2). Our method leads in all six categories, with the largest margin over PSDR-Room on sofas. Their broad fabric regions make missing or mismatched texture patterns particularly conspicuous.

4.5 Ablation Study

We ablate generated textures, view selection, staged optimization, and physics-based procedural retrieval on the chair and bag in Fig. 5. Variants within each row use the same reference, target mesh, and evaluation viewpoint.

W/o textures removes the generated-texture branch and optimizes only the retrieved procedural material. **W/o best-view** replaces the selected input view with a camera sampled once at random before optimization; its structure-aligned supervision image is generated for that camera. **W/o staged** jointly optimizes lighting, UV transforms, and material parameters for 200 iterations instead of using the 50/50/100-iteration schedule. **W/o physics-based procedural retrieval** initializes each segment with a randomly selected procedural material rather than a candidate ranked by the decomposed reflectance. All remaining components are retained, with 512×512 rendering and 64 samples per pixel.

Removing generated textures weakens the chair's upholstery stripes and the bag's ornamental detail. The other ablations change the appearance or placement of these patterns, whereas the full method more closely reproduces the salient material patterns in both references.

5 Conclusion

We presented **Style2Shape**, a framework for recovering editable, physically-based materials for a target 3D mesh from a single reference image. Our multi-stage design addresses the ill-posed nature of single-image material acquisition by (i) synthesizing a structure-aligned supervision image via structure-guided appearance transfer, (ii) initializing each segment with a hybrid of procedural reflectance and generated texture detail, and (iii) refining all parameters within a differentiable,



Figure 5: Qualitative ablation results for the chair and bag. For each target mesh, all variants are rendered from the same evaluation viewpoint.

physics-based rendering loop. The resulting assets are exported as standard PBR maps, enabling direct integration into existing graphics pipelines.

Editability and Downstream Use. The retained procedural parameters and UV transforms expose controls for revising color, pattern scale, orientation, and reflectance after fitting. A designer can test the recovered material under new lighting, adjust its procedural base, or replace the generated albedo while preserving the other PBR channels. This makes the result a reusable material asset rather than a single synthesized view. Appendix D shows the exported maps and their separate channels.

Limitations and Future Work. Our framework has several limitations. First, the input mesh is assumed to have usable UV coordinates and material groups, or to admit a reliable preprocessing segmentation. Incorrect groups merge regions that should have different reflectance, while very small or heavily occluded segments provide weak evidence for retrieval and optimization. Jointly reasoning about material boundaries and appearance would reduce this dependence on asset preparation.

Second, one image cannot observe the back side of an object or disambiguate all combinations of illumination and reflectance. For real photographs with complex, spatially varying lighting, intrinsic decomposition can be imperfect, and those errors may propagate to retrieval. Multiple reference views, uncertainty-aware decomposition, or regularization from measured material databases could make the recovered channels more stable.

Third, most main-paper experiments use references and targets from related semantic categories. The cross-category examples in Appendix A show that larger gaps are possible, but they do not exhaust the space of shape-style mismatch. Stronger structural priors and explicit correspondence confidence would be useful when a reference pattern has no clear placement on the target.

Fourth, the generated component currently modifies albedo only. This conservative design protects roughness, metallicity, and normals with a procedural prior, but it cannot reproduce reference-specific non-albedo detail that is missing from the selected graph. Extending the hybrid formulation to those channels will require cross-channel constraints so that generated variations remain physically coherent.

Finally, Stage 1 relies on a proprietary image editing model, and the present evaluation contains 30 user-study participants. Open editing alternatives, additional material databases, and tests with measured BRDF ground truth would improve reproducibility and support stronger claims about parameter accuracy.

AI USE STATEMENT

An image editing model was used as an explicit component of Style2Shape to generate structure-aligned supervision images and segment-wise albedo textures, as described in Sections 3.2 and 3.3. AI tools also assisted with language editing and polishing of the manuscript. The authors reviewed the AI-assisted outputs and take responsibility for the results and claims in this paper.

REFERENCES

Adobe Inc. Substance 3D Designer, 2024. URL https://www.adobe.com/products/substance3d-designer.html. Accessed: 2024.

Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 22246–22256, 2023.

Ye Fang, Zeyi Sun, Tong Wu, Jiaqi Wang, Ziwei Liu, Gordon Wetzstein, and Dahua Lin. Make-it-real: Unleashing large multimodal model for painting 3d objects with realistic materials. *Advances in Neural Information Processing Systems*, 37:99262–99298, 2024.

Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2414–2423, 2016.

Yiwei Hu, Paul Guerrero, Milos Hasan, Holly Rushmeier, and Valentin Deschaintre. Node graph optimization using differentiable proxies. In *ACM SIGGRAPH 2022 conference proceedings*, pp. 1–9, 2022.

Xin Huang, Tengfei Wang, Ziwei Liu, and Qing Wang. Material anything: Generating materials for any 3d object via diffusion. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 26556–26565, 2025.

Wenzel Jakob, Sébastien Speierer, Nicolas Roussel, Merlin Nimier-David, Delio Vicini, Tizian Zeltner, Baptiste Nicolet, Miguel Crespo, Vincent Leroy, and Ziyi Zhang. Mitsuba 3 renderer, 2022. https://mitsuba-renderer.org.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. Differentiable monte carlo ray tracing through edge sampling. *ACM Transactions on Graphics (TOG)*, 37(6):1–11, 2018.

Yuchen Li, Ujjwal Upadhyay, Habib Slim, Ahmed Abdelreheem, Arpit Prajapati, Suhail Pothigara, Peter Wonka, and Mohamed Elhoseiny. 3DCoMPaT: Composition of materials on parts of 3d things. In *17th European Conference on Computer Vision (ECCV)*, 2022.

Minghua Liu, Ruoxi Shi, Linghao Chen, Zhuoyang Zhang, Chao Xu, Xinyue Wei, Hansheng Chen, Chong Zeng, Jiayuan Gu, and Hao Su. One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10072–10083, 2024.

Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9298–9309, 2023.

Ivan Lopes, Fabio Pizzati, and Raoul de Charette. Material palette: Extraction of materials from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4379–4388, 2024.

Guillaume Loubet, Nicolas Holzschuch, and Wenzel Jakob. Reparameterizing discontinuous integrands for differentiable rendering. *ACM Transactions on Graphics (TOG)*, 38(6):1–14, 2019.

Oscar Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. Text2mesh: Text-driven neural stylization for meshes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13492–13502, 2022.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8280–8290, 2022.

Liang Shi, Beichen Li, Miloš Hašan, Kalyan Sunkavalli, Tamy Boubekeur, Radomir Mech, and Wojciech Matusik. Match: Differentiable material graphs for procedural material capture. *ACM Transactions on Graphics (TOG)*, 39(6):1–15, 2020.

George Tang, William Zhao, Logan Ford, David Benhaim, and Paul Zhang. Segment any mesh: Zero-shot mesh part segmentation via lifting segment anything 2 to 3d. *arXiv e-prints*, pp. arXiv–2408, 2024.

Trimble Inc. 3D Warehouse, 2024. URL `https://3dwarehouse.sketchup.com/`. Accessed: 2024.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. doi: 10.1109/TIP.2003.819861.

Xudong Xu, Zhaoyang Lyu, Xingang Pan, and Bo Dai. Matlaber: Material-aware text-to-3d via latent brdf auto-encoder. *arXiv preprint arXiv:2308.09278*, 2023.

Kai Yan, Fujun Luan, Miloš Hašan, Thibault Groueix, Valentin Deschaintre, and Shuang Zhao. Psdr-room: Single photo to scene using differentiable rendering. In *SIGGRAPH Asia 2023 Conference Papers*, pp. 1–11, 2023.

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 *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pp. 4347–4356, 2024.

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 2024 Conference Papers*, SIGGRAPH '24, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400705250.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018.

Shangzhan Zhang, Sida Peng, Tao Xu, Yuanbo Yang, Tianrun Chen, Nan Xue, Yujun Shen, Hujun Bao, Ruizhen Hu, and Xiaowei Zhou. Mapa: Text-driven photorealistic material painting for 3d shapes. In *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–12, 2024.

A Additional Results Gallery

This section presents additional qualitative results across object categories, reference styles, and target geometries.

Multi-Model, Multi-Category Variation. Figures 6 and 7 show transfers on tables, beds, sofas, and bags with varied target geometries and reference appearances.

Multiple Reference Styles on One Model. Figure 8 shows a single geometry paired with multiple reference images to produce different material appearances, including fabric, leather, and wood.

Cross-Category Transfers. Figure 9 shows cross-category transfers from a table reference to a chair and from a handbag reference to a table. The renderings retain color and pattern cues from the respective references on different target geometries.

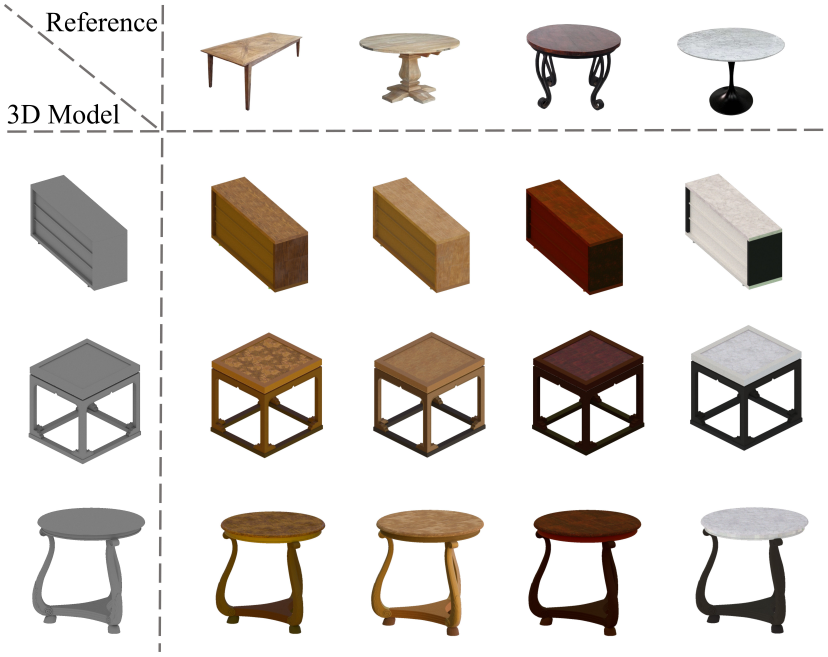

Figure 6: Material generation results within the tables category.

B Additional Pipeline Analysis

We visualize intermediate outputs and describe the role of structure-guided transfer and hybrid material initialization.

Structure-Guided Transfer. A central difficulty is the domain gap between a 2D reference image and a 3D target shape. Direct projection can distort or misalign reference patterns on the mesh. Stage 1 produces a structure-guided transfer image I_{tr} that follows the target geometry while incorporating reference material cues (Fig. 10). The resulting image provides pixel-aligned supervision for subsequent optimization.

Hybrid Material Components. Figure 11 visualizes the generated albedo textures $T_{A,k}^{\text{gen}}$ from Stage 2. The procedural component supplies the reflectance channels, while the generated texture contributes instance-specific albedo patterns. Both are retained in the hybrid initialization.



Figure 7: Material generation results on multiple categories (beds, sofas, bags) paired with varied reference materials.



Figure 8: Multiple material appearances transferred to the same target geometry from different reference images.



Figure 9: Cross-category material transfer from a table reference to a chair (left) and from a handbag reference to a table (right).

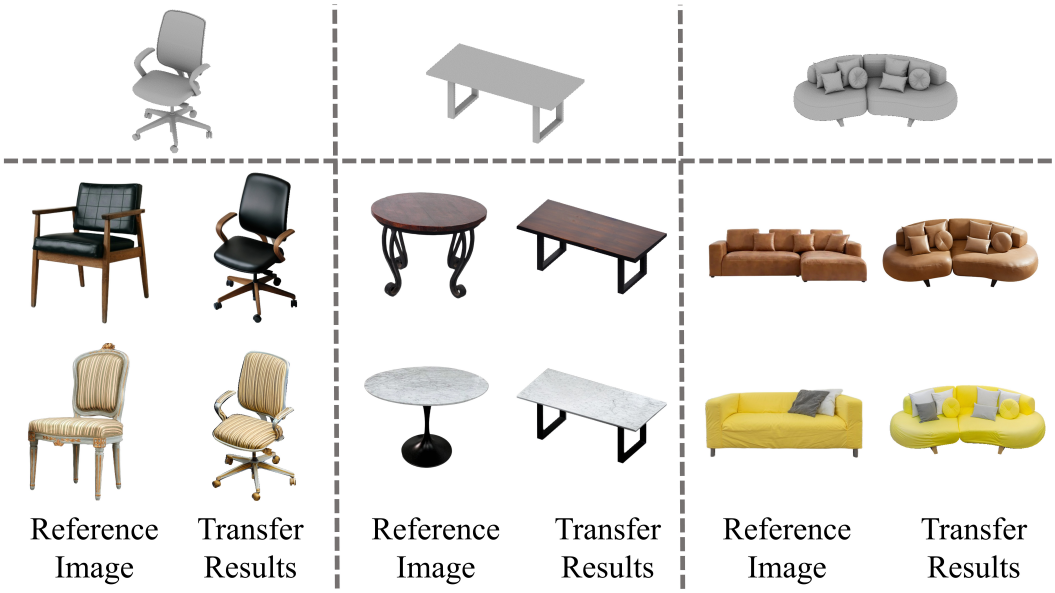

Figure 10: Structure-guided appearance transfer results (Stage 1). An image editing model transfers reference material cues onto the target geometry; the resulting images serve as supervision for material optimization.

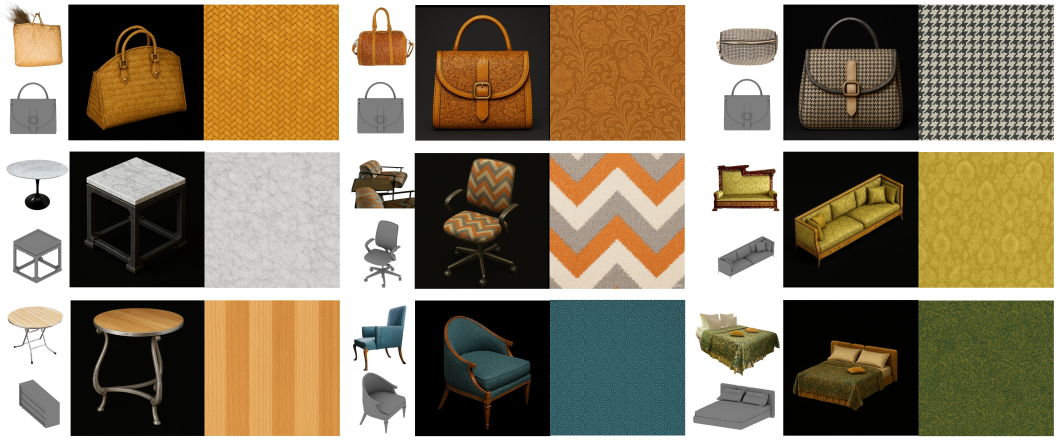

Figure 11: Visualization of pipeline intermediate results. Left: Structure-guided transfer images (I_{tr}) from Stage 1 providing pixel-aligned supervision. Right: Generated albedo textures ($T_{A,k}^{\text{gen}}$) from Stage 2 capturing detailed patterns for the hybrid material representation.

C Additional Method Details

Viewpoint Selection. The maximin term is important for multi-material objects. Maximizing only the number of visible segments can select a grazing view in which a small segment occupies too few pixels to provide reliable retrieval or perceptual supervision. By maximizing the least-visible segment area among views with maximal coverage, the selected camera favors evidence that is both diverse and usable. This remains a single-view compromise: completely occluded regions receive no direct image evidence and inherit their initialization from the corresponding material segment.

Generated Supervision. Masking excludes background content and cast shadows outside the object from the material objective. Because I_{tr} is generated, it serves as appearance evidence rather than measured ground truth. Procedural retrieval supplies a reflectance prior, and differentiable rendering constrains the final appearance through an explicit PBR model.

Image Editing Prompts. For Stage 1, the untextured target rendering is Image 1 and the reference photograph is Image 2. The prompt requires the model to "Preserve the geometry, silhouette, and spatial layout of Image 1," "Transfer the material appearance from Image 2, including color, texture, and reflectance cues," and "Avoid introducing new structures; apply style changes only to the visible surface appearance." For Stage 2, each masked material region is provided with the instructions "Minimize texture distortion to ensure proper tiling" and "Preserve the color and pattern characteristics."

Hybrid Texture and Retrieval. Matching in screen space accounts for the selected camera and segment visibility in the pixels used for optimization. Retrieval supplies the procedural base, while the generated texture represents reference patterns absent from the selected graph. Tileability is essential because a texture may repeat over large UV regions and across views; non-tileable crops can introduce seams visible after camera motion. The continuous weight w_k allows the optimizer to combine the procedural base pattern with the generated texture.

Staged Optimization. Optimizing all variables from the beginning is poorly conditioned. Lighting intensity can compensate for albedo, a change in texture scale can mimic a different procedural pattern, and an incorrect orientation can be partially absorbed by the blend weight. The staged schedule reduces these ambiguities by first explaining coarse radiance, then aligning spatial texture statistics, and only afterwards allowing the material representation to change jointly. Freezing the material during lighting estimation prevents the optimizer from baking illumination differences into the albedo. The discrete UV search provides a starting basin for periodic texture parameters; setting $w = 1$ during alignment exposes the generated pattern before the procedural component can dominate the VGG statistics. In final joint refinement, keeping the environment fixed prevents repeated exchanges of energy between illumination and material, while fine-tuning the UV transform adapts pattern placement to the changing blend.

Loss Construction. Because $D(\cdot)$ returns decomposed appearance channels, the reconstruction term constrains more than RGB color: it encourages agreement in the low-frequency albedo and reflectance cues used for retrieval. The segment-wise style term complements this global constraint. Gram matrices aggregate feature correlations and are therefore tolerant to small local misalignments while remaining sensitive to repeated texture statistics Gatys et al. (2016). Applying the term separately to each material segment prevents a visually dominant segment from explaining the texture statistics of smaller materials. The channel and feature weights control the relative contribution of material regions and feature scales.

For a VGG-16 feature map $F^l \in \mathbb{R}^{C_l \times N_l}$ at layer l, with $N_l = H_l W_l$, the Gram matrix is

$$\mathcal{G}^l_{ij} = \sum_{p=1}^{N_l} F^l_{ip} F^l_{jp}. \tag{6}$$

The per-segment style loss is the weighted squared Frobenius distance between the rendered and transferred feature Gram matrices over the selected layers $\mathcal{F}$:

$$\mathcal{L}^k_{\text{style}}(I_{\text{render},k}, I_{tr,k}) = \sum_{l \in \mathcal{F}} \frac{\omega_l}{C_l^2} \left\| \mathcal{G}_l(I_{\text{render},k}) - \mathcal{G}_l(I_{tr,k}) \right\|_F^2. \tag{7}$$

D Analysis of Generated PBR Materials

Recovered PBR Maps. Figure 12 shows the recovered PBR maps. The albedo maps combine procedural base color with generated texture detail, while roughness, metallic, and normal maps come from the optimized procedural component. The resulting maps retain separate, editable channels for standard PBR rendering.



Figure 12: Rendered results (left) and the corresponding albedo, roughness, metallic, and normal maps (right).

E Experimental Details

Dataset and Material Library. Our 3D model dataset is sourced from 3DCoMPaT Li et al. (2022) and 3D Warehouse Trimble Inc. (2024), covering six categories: bags, beds, buildings, chairs, tables, and sofas, with at least 5 models per category. Reference images consist of real-world photographs under CC0 license, similarly spanning all six categories with at least 5 images per category. For material matching and optimization, we collect 263 base procedural materials from Substance Designer Adobe Inc. (2024) in .sbs format. We sample 5 parameter variants for each base material by varying key parameters such as roughness, scale, and color, resulting in a total library of 1,315 material instances exported at 512×512 resolution.

Differentiable Renderer and Optimization. Our physics-based optimization pipeline is built upon the **Mitsuba 3** differentiable renderer Jakob et al. (2022). For propagating gradients through the rendering process, we utilize the **reparameterization** technique of Loubet et al. Loubet et al. (2019), which handles discontinuities from visibility events. We employ a **path tracing integrator** for all rendering operations. The rendering resolution is set to 512×512 pixels. To generate high-quality, anti-aliased images, we use an independent sampler with 64 samples per pixel (spp). We initialize the environment map E as a 256×256 HDR image with uniform unit radiance and apply $8\times$ downsampling during the rendering loss computation. The perceptual loss employs a pre-trained VGG-16 network. For UV transformation initialization, we evaluate discrete candidates with rotation angles $r \in \{0, \pi/4, \pi/2, -\pi/2, -\pi/4\}$ and scale factors $s \in \{0.5, 1.0, 2.0, 4.0, 8.0\}$. We employ a three-stage optimization schedule using the Adam optimizer Kingma & Ba (2014): (i) 50 iterations for lighting estimation with lr = 10^{-2}, where we apply a weak **total variation regularizer** to the environment map E to encourage spatial smoothness; (ii) 50 iterations for UV alignment with lr = 10^{-3}; and (iii) 100 iterations for joint refinement with cosine annealing from lr = 10^{-2} to 10^{-4} for material parameters. The complete pipeline requires approximately 10 minutes on a single NVIDIA RTX 3090 GPU.

Data Availability. The datasets and code used in this study are available upon reasonable request. The 3D models are sourced from publicly available datasets including 3DCoMPaT and 3D Warehouse. The procedural material library is based on materials from Substance Designer. Reference images are real-world photographs under CC0 license.