

Physics-Based Inverse Rendering using Combined Implicit and Explicit Geometries

Supplemental Materials

G. Cai^{1,2}, K. Yan^{1,2}, Z. Dong², I. Gkioulekas³, S. Zhao¹

¹University of California, Irvine

²Meta Reality Labs Research

³Carnegie Mellon University

Appendix A: Extra Experiments

We now show additional comparisons using both synthetic and real data.

Figures 1 and 2 are extended versions of Figure 3 of the main paper. In both figures, we show per-vertex distance visualizations by projecting each vertex on the reconstructed meshes to the ground-truth (marked as “*Recon. to GT*”) and vice versa (marked as “*GT to recon.*”). The IDR* baseline[†] has difficulties recovering the geometric details of the GT. The mesh-based baseline, on the other hand, manages to capture the geometric details locally but fails to obtain correct global topologies, resulting in missing or redundant geometric features. Our technique, by using both implicit and explicit geometries, enjoys the advantage of both representations.

In Figures 3–8, we compare reconstruction results with the mesh-based baseline. We note that IDR* is not applicable for these examples as they use environmental lighting that cannot be easily handled by the IDR framework (in a physics-based fashion). Thus, we mainly compare to the purely mesh-based method. All renderings in these are from novel views (that are not used for inverse-rendering optimizations). To quantitatively compare the qualities of these renderings, we compute their PSNR (shown under each rendering).

Appendix B: Social Impact

We do not foresee our technique to negatively impact our society in any significant fashion. Our research does not involve human subjects or human-derived data. Further, our main application—the digitization of 3D objects and scenes—is more about building virtual environments (or meta-verses) than manipulating the physical one, and has little to no environmental impact.

[†] This baseline is obtained by adopting the IDR technique. To compute the distance visualizations, we convert the resulting implicit shapes into meshes using high-resolution marching cube.

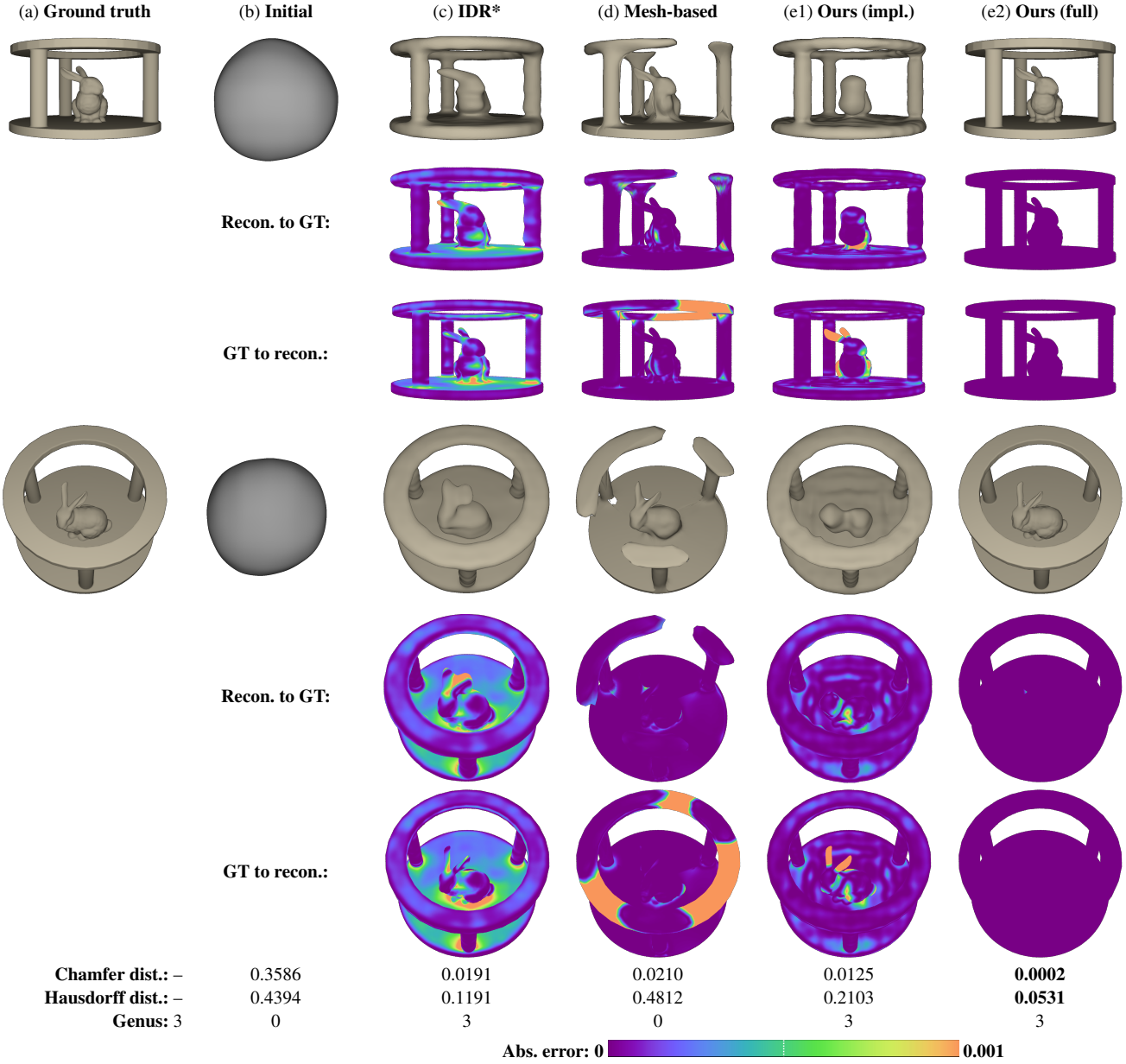


Figure 1: Inverse-rendering comparison (bunny temple): We show reconstruction results generated using IDR*—a modified version of IDR that uses physics-based shading—in (c), mesh-based optimization in (d), our implicit stage in (e1), and our full pipeline in (e2). All methods shared identical initializations shown in (b). The number below each reconstruction result indicates the Chamfer distance between the reconstructed and groundtruth geometries (normalized so that the GT has a unit bounding box).

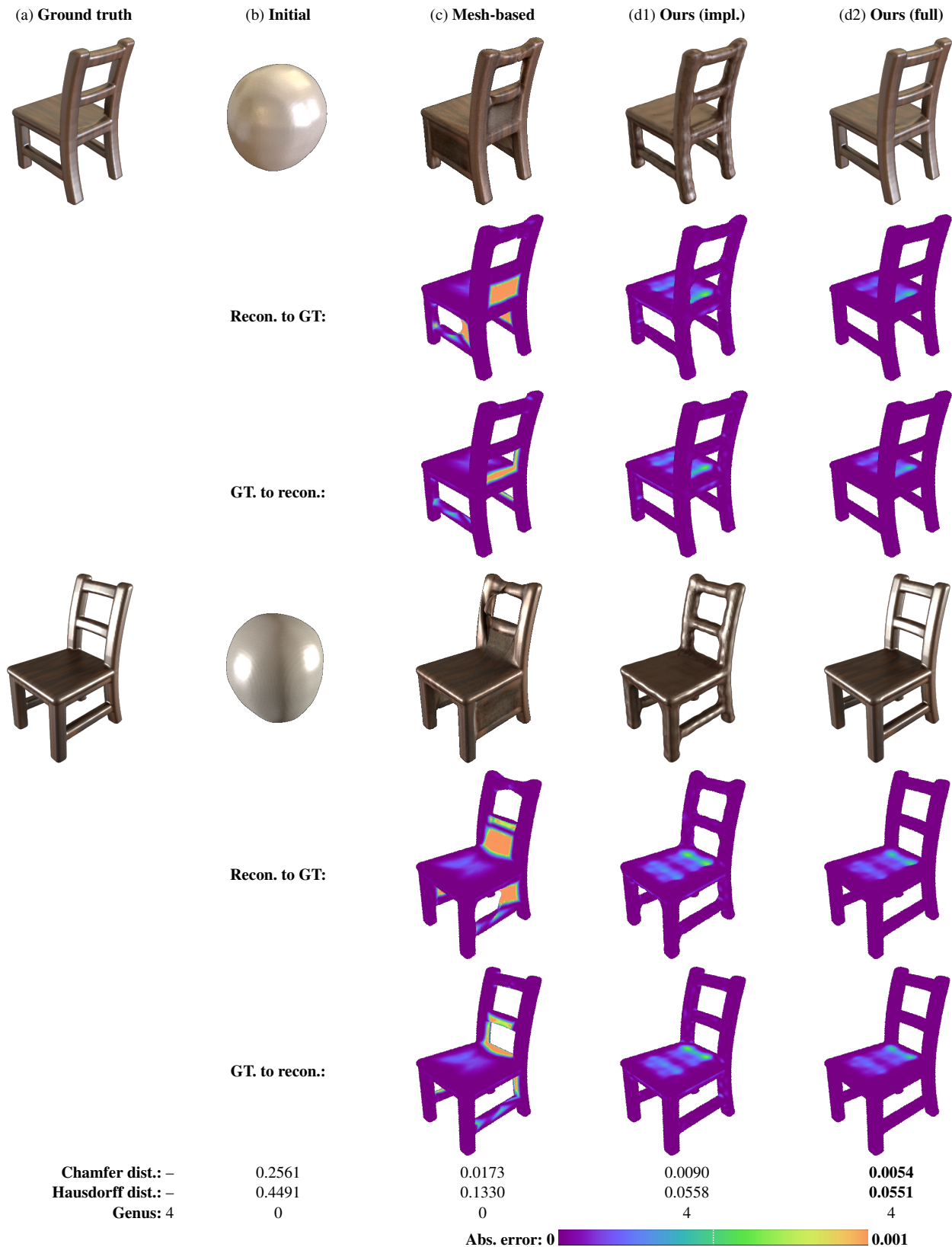


Figure 2: Inverse-rendering comparison (chair): We show reconstruction results generated using mesh-based optimization in (c), our implicit stage in (d1), and our full pipeline in (d2). All methods share identical initializations shown in (b).





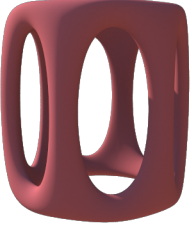

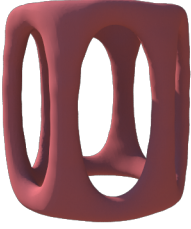
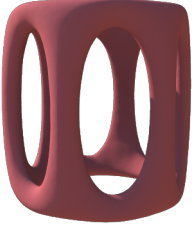
(a) Ground truth	(b) Mesh-based	(c1) Ours (impl.)	(c2) Ours (full)
			
PSNR: –	20.23	31.03	47.98
			
PSNR: –	23.44	30.99	45.45
Chamfer dist.: –	0.0661	0.0068	0.0008
Hausdorff dist.: –	0.4054	0.0222	0.0138
Genus: 7	0	7	7

Figure 3: Inverse-rendering comparison (frame): We show reconstruction results generated using mesh-based optimization in (b), our implicit stage in (c1), and our full pipeline in (c2). All methods share identical initializations similar to Figure 2-b.

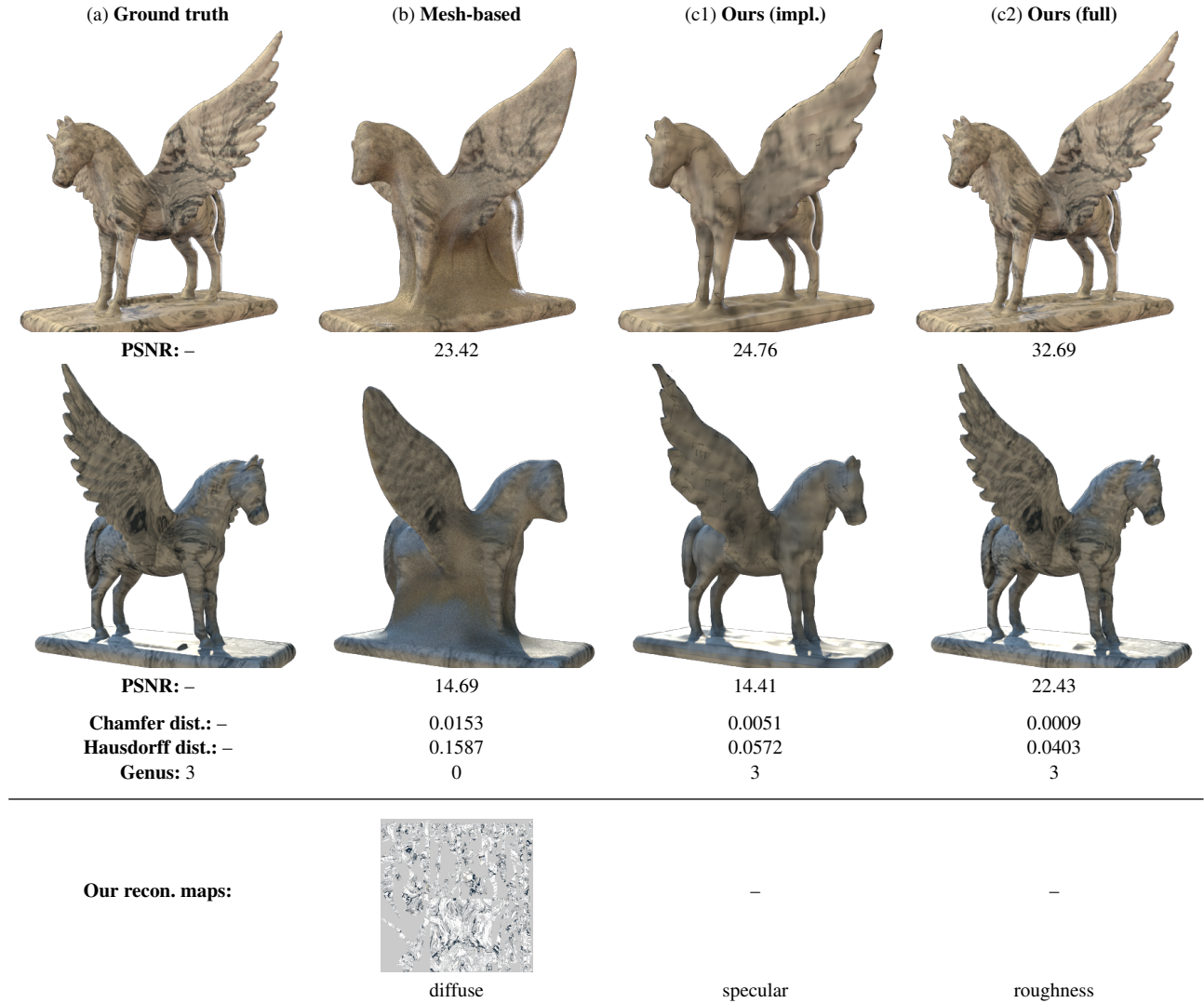


Figure 4: Inverse-rendering comparison (Pegasus): We show reconstruction results generated using mesh-based optimization in (b), our implicit stage in (c1), and our full pipeline in (c2). All methods share identical initializations similar to Figure 2-b.

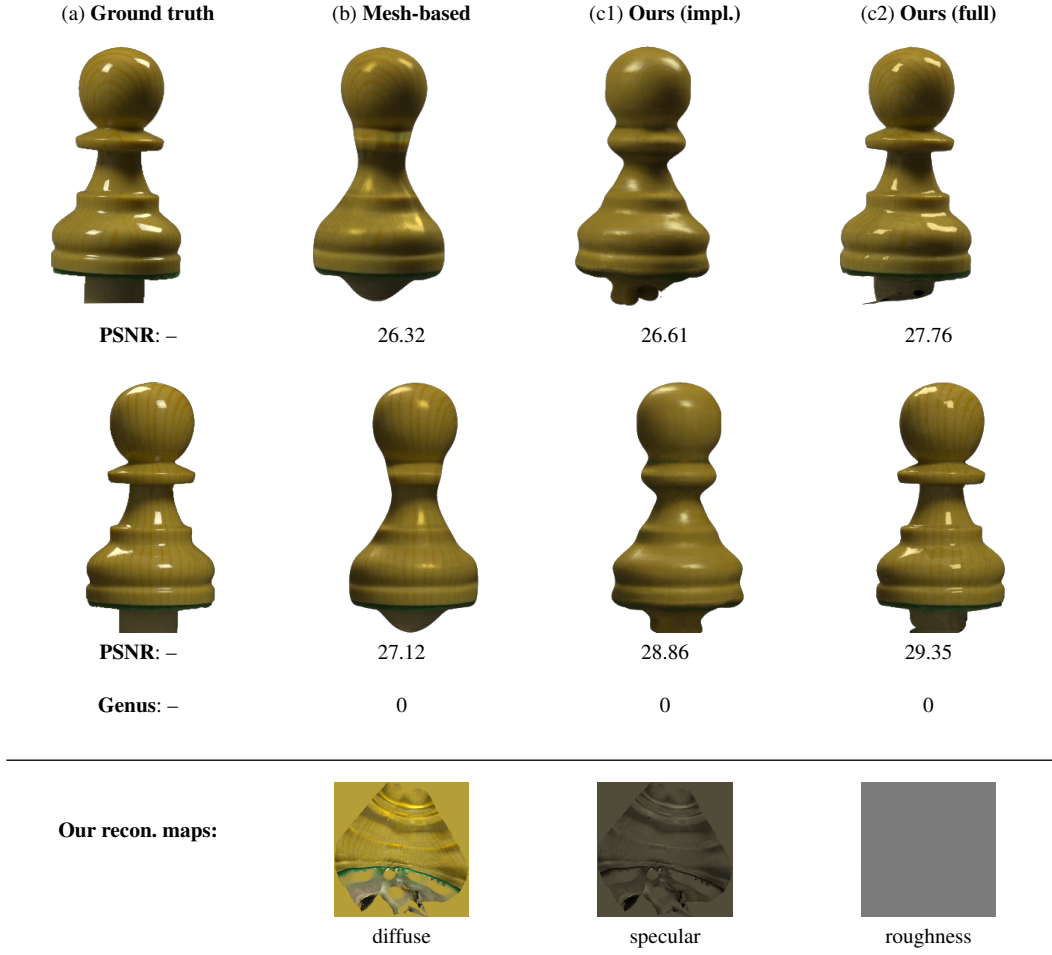


Figure 5: Inverse-rendering comparison (chess): We show reconstruction results generated using mesh-based optimization in (b), our implicit stage in (c1), and our full pipeline in (c2). All methods share identical initializations similar to Figure 2-b.

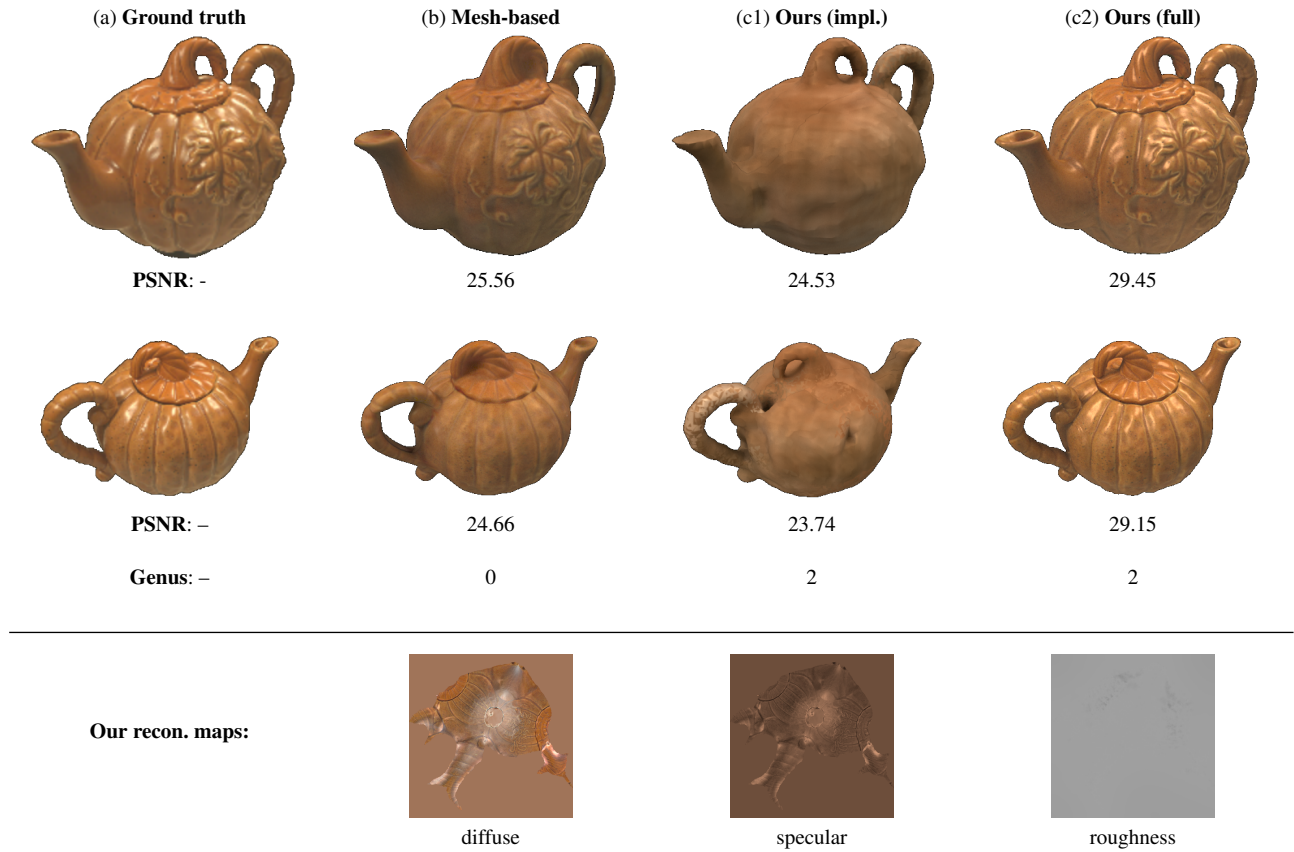


Figure 6: Inverse-rendering comparison (teapot): We show reconstruction results generated using mesh-based optimization in (b), our implicit stage in (c1), and our full pipeline in (c2). All methods share identical initializations similar to Figure 2-b.

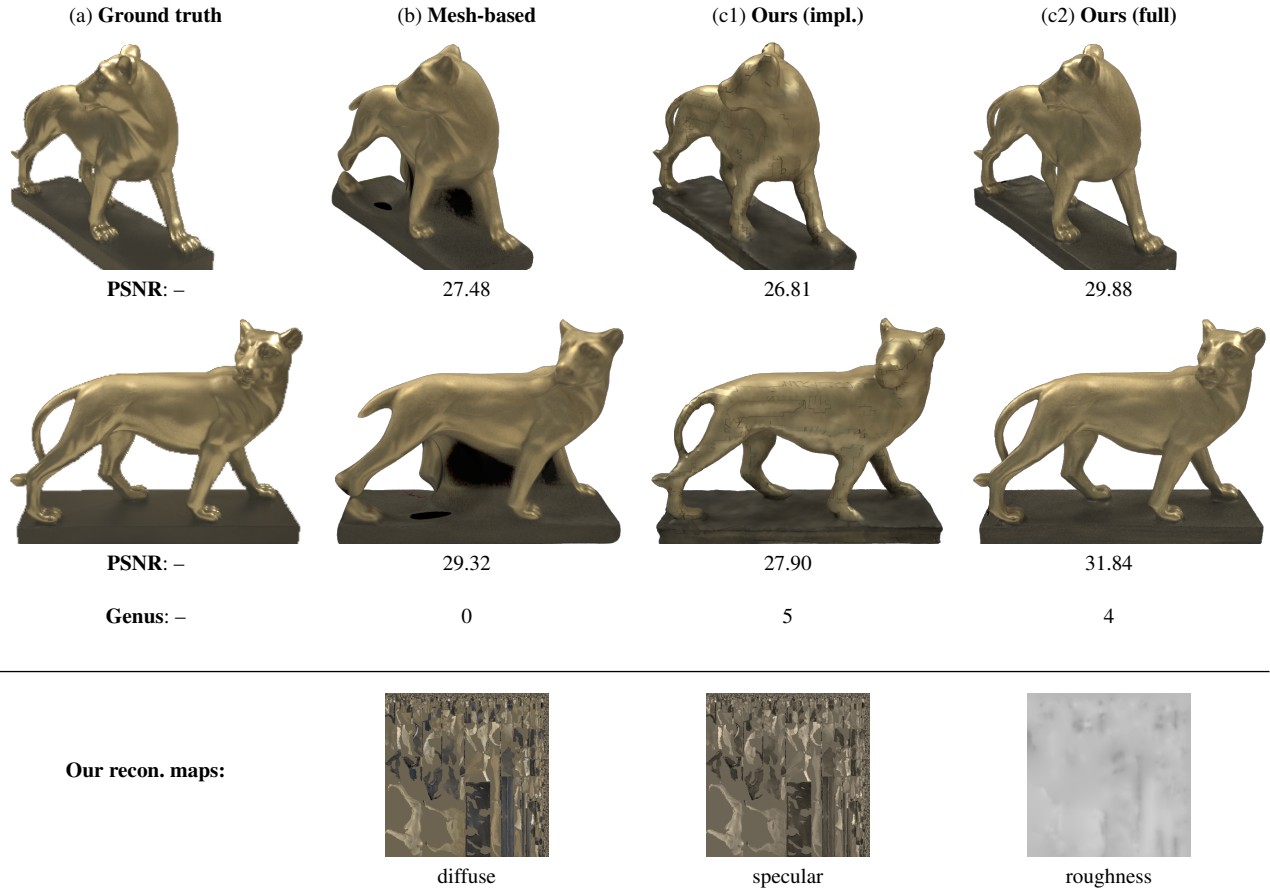


Figure 7: Inverse-rendering comparison (leopard): We show reconstruction results generated using mesh-based optimization in (b), our implicit stage in (c1), and our full pipeline in (c2). All methods share identical initializations similar to Figure 2-b.

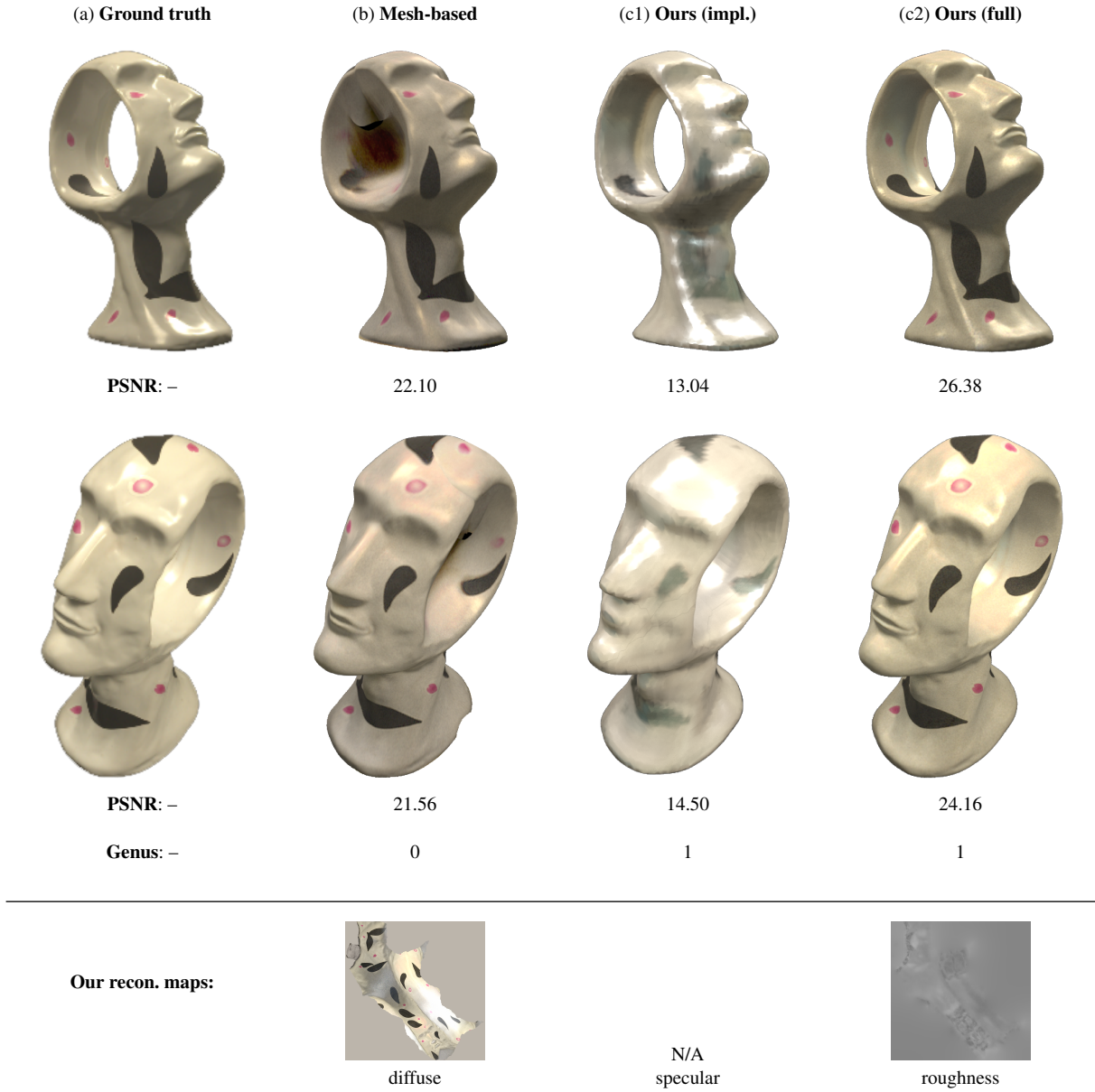


Figure 8: Inverse-rendering comparison (head): We show reconstruction results generated using mesh-based optimization in (b), our implicit stage in (c1), and our full pipeline in (c2). All methods share identical initializations similar to Figure 2-b.