On Neural BRDFs: A Thorough Comparison of State-of-the-Art Approaches
Supplementary Material

Florian Hofherr[1,2] Bjoern Haefner[†] [1,2,3] Daniel Cremers[1,2]

[1]Technical University of Munich [2]Munich Center for Machine Learning [3]NVIDIA

This supplementary material gives additional information on several aspects of our work. In Appendix A, we present more details on the neural BRDF modeling, including a discussion of the angles and an overview of the parametric models. In Appendix B, we share more insight on the training and the evaluation, including additional details on the loss formulation and the regularizers for the additive enhancement. Finally, we present several additional experiments in Appendix C, among them additional renderings of the objects and an analysis of the angle parametrization.

A. Additional Details on the Models

In this section, we give additional details on the neural BRDF modeling. Appendix A.1 reviews the Rusinkiewicz angles used to parametrize the view and light direction, Appendix A.2 gives an overview of the intrinsic encoding used to parametrize the neural BRDFs directly on the meshes. Finally, Appendix A.3 reviews the parametric models used to construct the parametric neural BRDFs.

A.1. Angle Definitions

While the BRDF is often defined in terms of the view and the light unit directions, v and l, respectively, those two quantities are elements on the (curved) $\mathbb{S}^2$ manifold, which makes them impractical to work with. Therefore, a parametrization in terms of the angles of the vectors is usually employed. The most obvious choice is to use the polar angles (θ_v, ϕ_v) and (θ_l, ϕ_l) of the two vectors directly, see Fig. A.1 on the left. We refer to this parametrization as *view-light angles*. For an isotropic BRDF, the absolute orientation in the tangential plane is irrelevant, and only the relative orientation of the vectors with respect to each other is important. In this case the triplet $(\theta_v, \theta_l, \phi_v - \phi_l)$ is sufficient to parameterize the BRDF.

However, as noted by Rusinkiewicz, the view-light angles are suboptimal to parameterize a BRDF [10]. They propose a novel parameterization that aligns important features of the BRDF, like the specular peak, with the coordinate axes. The representation is based on the half-vector

$$h = \frac{v+l}{\|v+l\|}. \tag{1}$$

and a "difference" vector d, which is obtained by rotating the light direction into a coordinate system, where the half vector h coincides with the normal, see Fig. A.1 on the right for an overview. We denote the surface normal by n and use an arbitrary tangential vector t and the binormal $b = n \times t$ to define a local coordinate system. The difference vector d is computed as

$$d = \mathrm{rot}_{b,-\theta_h} \mathrm{rot}_{n,-\phi_h} l, \tag{2}$$

where $\mathrm{rot}_{x,\alpha} y$ means the rotation of vector y around the axis x by the angle α. The full Rusinkiewicz angles read $(\theta_h, \phi_h, \theta_d, \phi_d)$. Isotropic BRDFs do not depend on ϕ_h; therefore, the triplet $(\theta_h, \theta_d, \phi_d)$ is sufficient for the parametrization in this case. See [10] for more details.

A.2. Neural Intrinsic Encoding

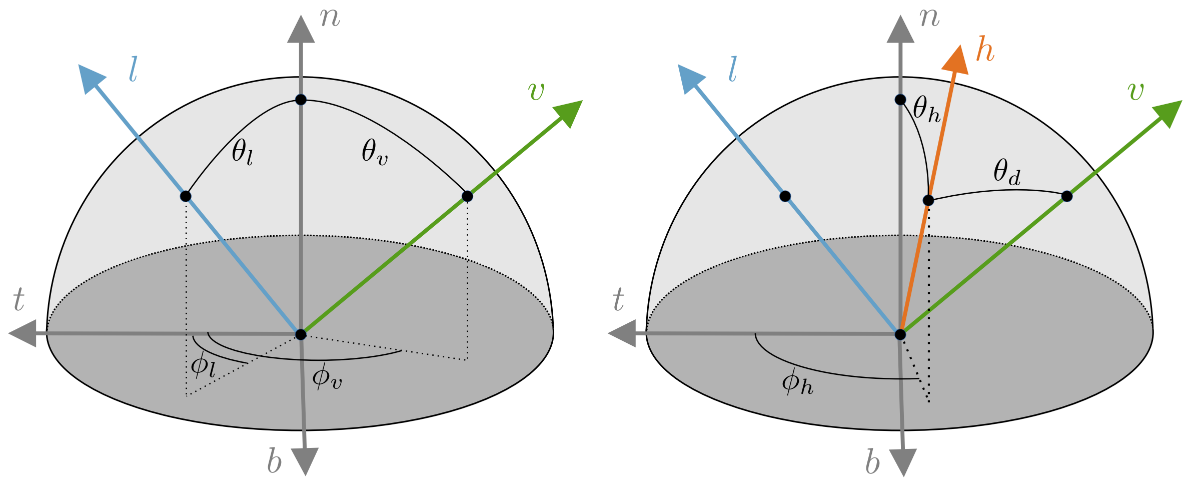
For the positional encoding on the manifold, we use the neural intrinsic encoding described by Koestler *et al.* [5], who propose to use a subset of the eigenfunctions of the Laplace-Beltrami operator (LBO)[1] for the encoding. Since the LBO is a generalization of the Laplace operator on an Euclidean domain, this form of encoding can be seen as an adaption of positional encoding from the Euclidean domain to manifolds like meshes. Given a closed, compact manifold $\mathcal{M} \subset \mathbb{R}^n$, we denote the eigenfunctions of the LBO on $\mathcal{M}$ by ζ_j. For the encoding, we consider a subset of indices $\mathcal{I} \subset \mathbb{N}$ and, given a point x on the mesh, use the encoding

$$\Theta_{LBO}(x) = (\zeta_j(x))_{j \in \mathcal{I}}. \tag{3}$$

In the discrete setting of a mesh, the values of the LBO eigenfunctions are computed on the vertices. We use barycentric interpolation to obtain the encoding for an arbitrary position on the mesh. In practice, it is not necessary

[†] The contribution was done while at TUM.

[1] We refer to [11] for more details on the LBO.



View-Light Angles

Rusinkiewicz Angles

Figure A.1. Visualization of the view-light and the Rusinkiewicz angles on the left and on the right, respectively. Shown are view and light direction, v and l in a local coordinate system given by surface normal n, a surface tangent t (which is arbitrary in the case of isotropic BRDFs) and the binormal $b = n \times t$. The view-light angles are simply the polar angles of v and l, while the Rusinkiewicz angles are given in terms of the half angle $h = \frac{v+l}{\|v+l\|}$ and a "difference" vector d which is the light direction l in a coordinate system with h as the normal.

to use a connected index set. We follow [5] and use several connected blocks of eigenfunctions for the encoding; see Appendix B.3 for the concrete numbers.

A.3. Parametric BRDF Models

In the following section, we give a short overview of the parametric models used in this work. We refer to the respective works of more details.

Realistic Phong Model We use the model described in [6]. To ensure energy conservation, we predict a combined value $k_{\text{full}} \in [0,1]^3$ for the reflectivities as well as a split percentage $\zeta \in [0,1]^3$ and compute the individual reflectivities as $k_d = \zeta \circ k_{\text{full}}$ and $k_s = (1-\zeta) \circ k_{\text{full}}$. To ensure the correct range, we use a sigmoid nonlinearity for k_{full} and ζ. The specular exponent $n \geq 1$ is computed using a softplus output function with an additive offset of 1.

Torrance-Sparrow Model For the specular part, we use the basic form of the microfacet model described in [16], which reads (omitting the dependence on the position)

$$f_{\text{spec}}(l, v) = \frac{D(h)F(v,h)G(l,v,h)}{4\langle n,l\rangle\langle n,v\rangle}, \tag{4}$$

where h is the half vector described in Eq. (1), D is the normal distribution function (NDF), F is the Fresnel term, G is the geometric shadowing term and $n \in \mathbb{S}^2$ is the surface normal in the local coordinate system. By $\langle n, l\rangle$ we denote the scalar product between n and l.

We use the (isotropic) Trowbridge-Reitz/GGX distribution [14, 17, 18] for D, which reads

$$D(h) = \frac{\alpha^2}{\pi(\langle n,h\rangle^2(\alpha^2-1)+1)^2} \tag{5}$$

with the roughness parameter $\alpha > 0$. For G, we use smiths method [14], which splits the term multiplicatively as

$$G(l,v,h) = \tilde{G}(l)\tilde{G}(v). \tag{6}$$

We use the Trowbridge-Reitz/GGX term [14, 17, 18] for $\tilde{G}$, which for a vector $w \in \mathbb{R}^3$ reads

$$\tilde{G}(w) = \frac{2\langle n,w\rangle}{\langle n,w\rangle + \sqrt{\alpha^2 + (1-\alpha^2)\langle n,w\rangle^2}}, \tag{7}$$

again with the same roughness parameter α. For F we use Schlick's Fresnel term [13], which reads

$$F(v,h) = F_0 + (1-F_0)(1-\langle v,h\rangle)^5, \tag{8}$$

with the characteristic specular reflectance $F_0 \in [0,1]^3$.

We combine the specular part with a Lambertian diffuse term, which we diminish by $1 - F$. The additional factor accounts for the fact that light reflected at the surface is not available for (diffuse) subsurface scattering. Hence, the full BRDF reads

$$f(l,v) = (1 - F(v,h))\frac{\rho_d}{\pi} + f_{\text{spec}}(l,v). \tag{9}$$

The parameters predicted by the neural network are the roughness parameter $r \in [0,1]$ from which we compute

$\alpha = r^2$, as well as the diffuse albedo $\rho_d \in [0,1]^3$ and the characteristic specular reflectance $F_0 \in [0,1]^3$. We use a sigmoid output nonlinearity for all of them to ensure the correct range. Note that we found the convergence to be more stable if we scale the input to the roughness nonlinearity by 0.5.

Fresnel Microfacet BRDF Model The Fresnel microfacet BRDF combines an extended microfacet specular term with a generalized radiometric body reflection [3]. We refer to the paper for an in-depth discussion of the method. One change in the approach for the specular term compared to the Torrance-Sparrow model is that a generalized normal distribution is used for D, which involves estimating a corresponding normalization constant during training. While we were able to predict the rest of the parameters in a spatially varying manner, we were unable to do so for the parameters of this generalized normal distribution since the estimation of the normalization makes the computation untraceable. Therefore, we estimate a single set of parameters for the distribution, while all the other parameters are estimated dependent on the position x. We want to point out again that for the synthetic data, this makes the estimation less complex since the material is uniform over the mesh.

Disney BRDF The Disney BRDF extends the microfacet model by several effects, and we refer to the paper for a detailed derivation and discussion [2]. The only adjustment we do compared to the full model is to fix the *anisotropic* parameter equal to 0, effectively making the BRDF isotropic. Since all of the other models are isotropic, this enables a meaningful comparison between the approaches. Note that we found the convergence to be more stable if we scale the input to the roughness nonlinearity by 0.5.

A.4. Softplus Scaling

While tuning the models, we occasionally observed runs for the additive purely neural models, where the training converged to a wrong local minimum. We found that scaling the softplus output of the specular MLP by a factor of 0.5 can stabilize the training in those cases. See also Appendix C.5 for a visualization of the failure case.

B. Training and Evaluation

This section contains additional details on the training and the evaluation. Appendix B.1 sheds more light on the gamma-corrected loss formulation. Appendix B.2 contains insight into the preprocessing of the DiLiGenT-MV dataset. Appendix B.3 presents the encoding parameters used for the training. Appendix B.4 gives details on the regularizers employed for the enhancement of the additional split. The excluded values for the computation of the RMSE are discussed in Appendix B.5. Finally, Appendix B.6 presents details on the Monte Carlo integration for the experiments on the energy conservation.

B.1. Loss Formulation

To avoid a dominant influence of the bright regions on the loss, we use a gamma mapping to transform the RGB values from linear to sRGB space, as described in Eq. (10) the main paper. We repeat the loss formulation here for convenience.

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}(\gamma(L_o(x,v)) - \gamma(L_{GT}(x,v))^2 \tag{10}$$

We use the following standard formula for the gamma mapping $g : [0,1] \to [0,1]$, $c_{\text{lin}} \mapsto c_{\text{sRGB}}$ [1]:

$$g(c_{\text{lin}}) = \begin{cases} \frac{323}{25}\, c_{\text{lin}} & \text{if } c_{\text{lin}} \leq 0.0031308 \\ \frac{211}{200}\, c_{\text{lin}}^{\frac{5}{12}} - \frac{11}{200} & \text{else} \end{cases} \tag{11}$$

B.2. Pre-Processing of the Data

The triangle meshes supplied with the DiLiGenT-MV dataset [8] have an unreasonably high number of vertices, which causes the computation of the LBO eigenfunctions to take very long. Therefore, we reduce the number of vertices from roughly 10^6 to about 10^5. To stay consistent with the simplified mesh, we compute the normals on the mesh rather than using the normal maps included in the dataset. To speed up the training process, we pre-compute and store the LBO eigenfunctions as well as the ray-mesh intersections and the shadow rays.

B.3. Encoding Parameters

For the eigenfunctions of the LBO we use 6 blocks between the 1st and the 512th eigenfunction. We use 64 eigenfunctions for the first block and follow up with 6 evenly spaced blocks of 16 eigenfunctions. For the positional encoding of the angles, we use 3 encoding frequencies.

B.4. Regularizers for the Enhanced Splitting

While the enhancement described in Sec. 4.4 in the main text shows improvements for both additive architectures, it also introduces additional ambiguity. The weight $\xi(x,l,v)$ makes the diffuse summand dependent on the directions, and therefore, this term can now potentially represent specular behavior as well. To separate the phenomena, we introduce two additional regularizers.

First, we employ an L1 loss between the raw diffuse part (without the additional weight ξ) and the ground truth. Again, we use the gamma mapping described in Appendix B.1.

$$\mathcal{L}_{\text{reg,diff}} = \frac{1}{N}\sum_{i=1}^{N} \|\gamma(f_{\text{d}}(x)) - \gamma(L_{GT}(x,v))\|_1 \tag{12}$$

The idea is to encourage the model to represent as much of the appearance as possible by the diffuse term. This will be limited to the non-view-dependent parts of the appearance automatically since the raw diffuse part $f_{\text{d}}(x)$ is not view-dependent.

Second, we employ an L1 regularizer on the specular part.

$$\mathcal{L}_{\text{reg,spec}} = \frac{1}{N}\sum_{i=1}^{N} \|f_{\text{s}}(x,l,v)\|_1 \tag{13}$$

The idea for this term is to encourage the model to represent only those components of the appearance that are actually view-dependent by the specular part. We use a weight of $5 \cdot 10^{-4}$ for both terms before adding them to the total loss.

Interestingly, the regularizers do not significantly change the reconstruction quality of the enhanced split; the quantitative evaluation with and without them is almost identical. However, they force a more reasonable split between the diffuse and specular parts. Without them, the model tends to predict extreme mixing colors on the different parts, that combined yield the correct color. The regularizers ensure that realistic results for the diffuse albedo and the specularities are obtained.

B.5. BRDF-Spcae Metric RMSE $\sqrt[3]{}$

For the semi-synthetic dataset, we utilize the availability of ground truth BRDF data to report a BRDF-space metric on the reconstruction quality. We follow the analysis of Lavoué *et al.* who have investigated a wide range of metrics and analyzed the correlation with reconstruction quality perceived by humans [7]. Their results show that applying the cubic root to the BRDF values before computing a standard *root mean squared error* (RMSE) between the reconstruction and the ground truth yields a metric that correlates well with human perception. We refer to this metric as RMSE $\sqrt[3]{}$.

Another aspect that they found helpful to increase the correlation is to discard BRDF values for grazing angles above 80°, which we also adopt in our analysis. A further reason to discard these values is the observation of Burley, who reported anomalies and extrapolation in the MERL data in that range of angles [2], which the semi-synthetic data inevitably inherits. Moreover, we reject values for saturated pixels since in this case, the image value was clipped during the data creation, and therefore the ground truth value is not a reliable reference. Note that BRDF-space metrics can only be reported for the semi-synthetic data, since the DiLiGenT-MV dataset does not contain ground truth BRDF values.

B.6. Monte Carlo Integration

To approximate the integral in the energy conservation (Eq. (4) in the main text) we use Monte Carlo integration with samples from the cosine-weighted hemisphere sampling. The approximation of the integral reads

$$\int_{\mathbb{H}} f(x,l,v)\cos\theta_v \,\mathrm{d}v \approx \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} \frac{f(x,l,v_i)\cos\theta_{v_i}}{p(v_i)}, \tag{14}$$

where x and l are the position and the light direction for which the integral is evaluated. $v_i \sim p(v_i)$ are the sampled view directions, which are drawn according to the cosine-weighted distribution on the hemisphere, which reads

$$p(v_i) = \frac{1}{\pi}\cos\theta_{v_i}. \tag{15}$$

Hence, the approximation in Eq. (14) can be simplified to

$$\int_{\mathbb{S}^2} f(x,l,v)\cos\theta_v \,\mathrm{d}v \approx \frac{\pi}{N_{MC}} \sum_{i=1}^{N_{MC}} f(x,l,v_i). \tag{16}$$

To ensure convergence, we sample $N_{MC} = 20k$ view directions for each randomly sampled point-light pair.

C. Additional Experimental Results

In this section, we present several additional results. Appendix C.1 and Appendix C.2 give further results and details for the experiments on the reciprocity approach and the number of layers for the directions from Sec. 6.2 in the main text. In Appendix C.3, we analyze the influence of the angle parametrization on the reconstruction quality. Appendix C.4 justifies our changes to the NeRFactor architecture, which is the basis of our additive separate architecture. Appendix C.5 gives insight to failure cases for the purely neural additive approaches that we observed occasionally. While Appendix C.6 presents a qualitative analysis of the diffuse and specular parts of the models based on a split of the reflectance Appendix C.7 uses the albedo maps to analyze how well the models can represent the spatially uniform BRDFs of the semi-synthetic dataset. Finally, Appendix C.8 provides additional qualitative and quantitative results for both datasets, comparing the various neural BRDF approaches.

C.1. Quantitative Results for the Reciprocity Approaches

As shown in Sec. 6.2 in the main text, the random input swap of view and light direction applied during training, as described in LitNerf [12], can reduce the RMSE of the reciprocity constraint. However, this strategy does not provide

	MERL [9]					DiLiGenT-MV [8]			
	ΔRMSE $\sqrt[3]{\ }$	ΔPSNR	ΔDSSIM	ΔLPIPS	ΔꟻLIP	ΔPSNR	ΔDSSIM	ΔLPIPS	ΔꟻLIP
Single MLP (rnd. in-swap)	-0.08	+0.04	+0.02	+0.02	+0.02	+0.06	-0.01	-0.01	-0.02
Single MLP (ours)	-0.08	-0.10	+0.02	+0.01	+0.03	-0.30	+0.03	+0.06	+0.07
Additive Separate (rnd. in-swap)	-0.07	+0.04	+0.02	+0.01	+0.02	0.00	0.00	0.00	+0.01
Additive Separate (ours)	-0.07	-0.05	+0.02	+0.02	+0.04	-0.02	0.00	+0.01	+0.01
Additive Shared (rnd. in-swap)	-0.05	-0.10	+0.04	+0.16	-0.12	-0.15	+0.01	+0.01	+0.01
Additive Shared (ours)	-0.01	-0.21	+0.06	+0.24	-0.05	-0.13	0.00	0.00	+0.01

Table C.1. Effect of the reciprocity strategies on the reconstruction quality. Shown are the differences to the results in Tab. 1 in the main text, which reports the reconstruction quality of the architectures without any reciprocity strategy. With a few exceptions, we see that the influence of both strategies on the results is fairly marginal, with a tendency for slightly worse results. RMSE $\sqrt[3]{\ }$, DSSIM, LPIPS and ꟻLIP are scaled by 100.

a guarantee that the reciprocity is actually fulfilled. In contrast, our input mapping, as proposed in Sec. 4.3, ensures that the reciprocity constraint is exactly fulfilled by construction. In Tab. C.1, we analyze the effect of both strategies on the reconstruction quality by comparing against the results without any reciprocity strategy. Overall, we observe only a minimal difference with a minor tendency for slightly worse results as a trade-off for the fulfilled reciprocity. One exception is the single MLP architecture, where, for the real-world examples, the random input swap seems to have a more noticeable advantage over our strategy.

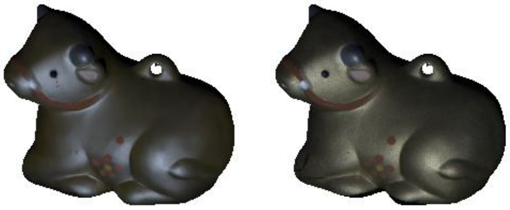
C.2. Architecture Changes Experiments: Number of Layers for View/Light Direction

In Sec. 6.2 in the main text, we analyzed the influence of the number of layers (NOL) for the directions on the reconstruction quality. In the following, we provide detailed architectural changes for the different models.

- For the single MLP architecture, we feed the directions in at layer 4 instead of 1, decreasing the NOL for the directions from 6 to 3.
- For the additive separate architecture, we reduce the NOL of the specular MLP from 4 to 2. Also, we remove the input skip.
- For the additive shared architecture, we reduce the NOL of the shared MLP from 5 to 3 while increasing the NOL of the diffuse and specular MLPs (and therefore NOL for the directions) by 2, respectively.

C.3. Influence of the Angle Parametrization

Following previous work [15, 19], we use the Rusinkiewicz angles [10] to parametrize the directions as inputs for the MLPs; see Appendix A.1 for a review and discussion. As noted by Rusinkiewicz, this parametrization aligns the specular peaks better with the coordinate axes, which benefits learning highly specular materials. The results in Tab. C.2 confirm that using the view-light angles as parametrization for purely neural BRDF models reduces the reconstruction quality compared to the Rusinkiewicz angles. Moreover, we see that, indeed, this effect is more prominent for the MERL-based semi-synthetic dataset, which contains more highly specular materials.



Scalar Specularity GT

Figure C.1. Qualitative BRDF reconstruction for the cow object from the DiLiGenT-MV dataset [8] for the additive separate architecture with a scalar specular term (as suggested in NeRFactor [19]). The results show, that a scalar specular term is unable to reconstruct the reflectance of this object and creates a spurious glow. This indicates that for some materials, a specular term with 3 channels is necessary to yield high-quality reconstructions.

C.4. Changes from the NeRFactor Architecture

While our additive separate architecture is based on NeRFactor [19], we made two changes, which improved the results for our data significantly. In this section, we present the comparison to justify these adjustments.

As a first change, we remove the albedo clamping. The original work empirically constrains the diffuse reflection (*i.e.* the albedo) to $[0.03, 0.8]$. Since we work in linear space, we transform these values from sRGB to linear space, which yields the range $[0.0023, 0.6038]$. The results in Tab. C.3 show that we obtain significantly worse results for the semi-synthetic MERL-based data. The reason lies in the metallic materials contained in this dataset. Metals show almost no subsurface scattering due to the free electrons [1]. As demonstrated in Appendix C.6 and visible in particular in Fig. C.5, all models replicate this behavior and show almost no diffuse contribution. However, the lower bound on the diffuse part imposed by the albedo clamping prohibits a negligible contribution of the albedo, and we observe a dark gray base color for the diffuse renderings. This leads

	MERL [9]					DiLiGenT-MV [8]			
	ΔRMSE ∛	ΔPSNR	ΔDSSIM	ΔLPIPS	ΔꟻLIP	ΔPSNR	ΔDSSIM	ΔLPIPS	ΔꟻLIP
Single MLP (view-light)	+0.11	-0.86	+0.05	+0.14	+0.15	-0.10	+0.01	+0.01	+0.01
Additive Separate (view-light)	+0.11	-0.85	+0.04	+0.12	+0.16	-0.03	+0.01	-0.02	+0.01
Additive Shared (view-light)	+0.17	-1.01	+0.05	+0.09	+0.07	-0.07	+0.01	-0.01	0.00

Table C.2. Effect of using the view-light angles instead of the Rusinkiewicz angles as a parametrization of the directions. Shown are the differences to the results in Tab. 1 in the main text, which reports the reconstruction quality of the architectures with the Rusinkiewicz angles. The results show that overall, the view-light parametrization reduces the reduction quality compared to the Rusinkiewicz parametrization. The difference is more significant for the semi-synthetic MERL-based dataset. The reason is that this data contains more highly specular materials with complex reflective patterns. The alignment of the specular peaks with the coordinate axes provided by the Rusinkiewicz angles seems to provide a significant benefit in this case. RMSE ∛, DSSIM, LPIPS and ꟻLIP are scaled by 100.

	MERL [9]					DiLiGenT-MV [8]			
	ΔRMSE ∛	ΔPSNR	ΔDSSIM	ΔLPIPS	ΔꟻLIP	ΔPSNR	ΔDSSIM	ΔLPIPS	ΔꟻLIP
Additive Separate (albedo clamp.)	+0.90	-2.25	+0.54	+0.84	+0.90	+0.05	0.00	-0.02	+0.01
Additive Separate (scalar spec.)	+0.41	-3.32	+0.10	+0.34	+0.74	-0.54	+0.01	+0.04	+0.32

Table C.3. Effect of using the albedo clamping and a scalar specular term proposed in NeRFactor [19] for the *additive separate* architecture. Shown are the differences to the results in Tab. 1 in the main text, which reports the reconstruction quality of the additive separate model with neither of the two. We see, that the albedo clamping reduces the reconstruction quality, in particular for the MERL-based data. The clamping prohibits the model from predicting an albedo close to zero, which is necessary, however, for the metallic materials contained in this dataset. See also Appendix C.6 and in particular in Fig. C.5. Similarly, the scalar specular term reduces the reconstruction quality for both datasets. We find, that for certain materials, an RGB specularity is necessary for a faithful reconstruction, see Fig. C.1. RMSE ∛, DSSIM, LPIPS and ꟻLIP are scaled by 100.

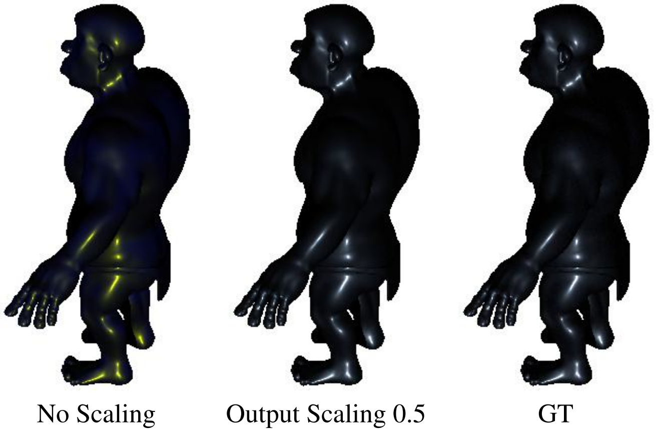


Figure C.2. Failure case for the additive shared architecture. For additive purely neural methods with softplus output nonlinearity, we observed occasional failures like this for very shiny materials. As described in Appendix A.4, scaling the output of the softplus function solves this issue.

to the decrease in construction quality. While we see slight improvements with the albedo clamping for non-metal materials and the real-world data, we still decided to remove it due to the significant performance decrease for metallic materials.

As a second change, we use an RGB specular term instead of a scalar one. While the original work assumes, that all color information can be handeled by the albedo network, we find, that for certain materials, a colored specular part is necessary for good reconstructions. The most extreme example we observed is the cow object from the DiLiGenT-MV dataset [8] as shown in Fig. C.1. We can clearly see, that an approach with a scalar specular term yields an inaccurate glow effect that is not observed for models with an RGB specular term (*cf*. the rendering for the additive separate architecture in Fig. 3 in the main text and Fig. C.3 in the appendix). While the effect is less prominent for other materials, Tab. C.3 confirms that a scalar specular term instead of an RGB one yields systematically worse results for both datasets. Recall that we observe a similar effect on the cow for the FMBRDF model [3], which also employs a scalar specular term; *cf*. Sec. 6.1 in the main text.

C.5. Failure Cases

Although all methods show quite stable convergence with the chosen parameters, we observed occasional issues with very shiny materials for the additive purely neural models that employ a softplus output nonlinearity. Fig. C.2 shows a failure example for the additive shared architecture. As described in Appendix A.4, scaling the output of the softplus function solves this issue.

C.6. Qualitative Results Diffuse and Specular Split

In Figs. C.3 to C.5, we present a quantitative analysis of the diffuse and the specular component of all models that split the BRDF into those two contributions. Overall, the methods mostly show a reasonable split. For the purely neural methods, we notice a tendency to represent a larger fraction of the appearance by the specular part, leading to darker diffuse parts. Our enhancements introduced in Sec. 4.4 and in particular the regularizers discussed in Sec. 4.4 and Appendix B.4 seem to improve on the disentanglement of diffuse and specular components. Fig. C.5 reveals that all models can represent the behavior of metallic objects, where almost no subsurface scattering is present, and indeed predict almost no albedo component.

C.7. Spatial Variation of the Reconstructed BRDFs

As described in the main text, the MERL BRDFs are uniform over the respective meshes for the semi-synthetic dataset. To assess how well the models can capture this spatial uniformity, we render the albedo without the cosine term for all models, that employ an additive split. The results for five materials in Fig. C.6 reveal very little spatial variation of the albedos, which indicates that all models are able to capture the spatial uniformity quite well.

C.8. Additional Comparison Results

We report quantitative results for the individual objects of the DiLiGenT-MV dataset in Tab. C.4 and qualitative results for both datasets in Figs. C.7 to C.13.

Real-World Data The quantitative evaluation on the individual objects in Tab. C.4 confirms that for the DiLiGenT-MV dataset [8], the difference between the approaches based on parametric models and purely neural methods is quite small. We even observe that for individual objects like the bear, some parametric approaches perform slightly better than the purely neural approaches – in particular, better than approaches with more layers for the directions (*e.g.* Single MLP). This behavior can also be observed qualitatively in Fig. C.7. The reason might be the noise in the real-world data, to which the purely neural methods seem to be slightly more sensitive. The fact that neural approaches with fewer layers for the directions (*e.g.* Additive shared) yield better results supports this claim, in particular in light of the analysis of the number of layers for the directions presented in Sec. 6.2.

Moreover, Fig. C.7 reveals that all methods show errors in similar regions - recesses in particular. We hypothesize un-modelled interreflections as a potential reason. Due to the indicator function in the rendering equation for our scenario (Eq. (9) in the main paper), shadows in the recesses will be completely black in our renderings. That is, however, a simplification because, in reality, some light reflected off the near surfaces will reach the shaded regions in the recesses. Therefore, larger errors for the image-based metrics in those parts of the mesh are expected.

Finally, Tab. C.4 confirms that for the novel additive strategy, which we proposed in Sec. 4.4, we observe consistent improvements in the extended vanilla additive models.

Semi-Synthetic Data Figs. C.8 to C.13 show a systematic advantage of purely neural methods for the challenging materials of the MERL dataset. The error maps reveal that in particular the specular peaks are much better represented, often showing a significant improvement over the parametric methods. While the difference is smaller for more diffuse materials, we still see an advantage of the purely neural methods.

Datasets	Error metric	RP [6]	TS [16]	FMBRDF [3]	Disney [2]	Single MLP	Add Sep	Add Shared	Add Sep (enh.)	Add Shared (enh.)
Bear	PSNR ↑	43.99	44.24	44.24	44.35	44.09	43.96	44.66	44.34	**44.74**
	DSSIM ↓	0.476	0.464	0.464	0.454	0.489	0.477	0.445	0.463	**0.442**
	LPIPS ↓	0.993	0.985	**0.973**	1.022	1.124	1.115	1.030	1.062	1.022
	ꟻLIP ↓	2.776	2.729	2.759	2.708	2.661	2.695	2.581	2.613	**2.550**
Buddha	PSNR ↑	36.30	36.41	36.50	36.35	35.50	35.96	**36.77**	36.07	36.71
	DSSIM ↓	1.298	1.275	1.257	1.281	1.387	1.299	1.228	1.283	**1.224**
	LPIPS ↓	2.278	2.272	2.240	2.376	2.489	2.257	2.211	2.229	**2.169**
	ꟻLIP ↓	4.212	4.187	4.184	4.243	4.156	4.051	3.945	4.014	**3.928**
Cow	PSNR ↑	46.02	46.57	43.84	46.57	46.22	46.37	47.08	46.49	**47.12**
	DSSIM ↓	0.395	0.373	0.422	0.372	0.391	0.381	0.358	0.374	**0.354**
	LPIPS ↓	1.682	1.686	1.885	1.701	1.635	1.593	1.595	1.570	**1.568**
	ꟻLIP ↓	2.270	2.133	3.066	2.159	1.987	1.969	1.933	1.961	**1.919**
Pot2	PSNR ↑	46.13	46.35	46.47	46.49	46.64	46.63	**46.92**	46.66	46.92
	DSSIM ↓	0.524	0.502	0.502	0.490	0.492	0.492	0.474	0.486	**0.472**
	LPIPS ↓	1.117	1.109	1.101	1.132	1.068	1.067	1.073	**1.054**	1.073
	ꟻLIP ↓	2.841	2.774	2.748	2.759	2.606	2.622	2.576	2.608	**2.569**
Reading	PSNR ↑	35.51	35.58	36.05	35.72	35.77	35.94	36.31	35.84	**36.38**
	DSSIM ↓	1.184	1.173	1.114	1.155	1.185	1.139	1.066	1.116	**1.052**
	LPIPS ↓	2.786	2.765	2.708	2.709	2.506	2.476	2.447	**2.403**	2.466
	ꟻLIP ↓	3.497	3.446	3.383	3.411	3.397	3.281	3.243	3.218	**3.172**

Table C.4. Quantitative comparison of the BRDF models on all individual objects of the real-world data [8]. PSNR, DSSIM and LPIPS are computed for the sRGB renderings. All quantities are first averaged over one object and then averaged over all objects in the respective dataset. DSSIM, LPIPS and ꟻLIP are scaled by 100. We see that for this dataset, the approaches based on parametric models (🟨) yield results that are comparable to the purely neural approaches (🟦). For individual objects, some parametric approaches even show slightly better results than the purely neural ones, in particular better than approaches with more layers for the directions (*e.g*. Single MLP). Again, a potential reason might be noise in the real-world data to which the purely neural models seem to be a little more sensitive; especially with more layers for the directions. This is supported by the observation that among the purely neural approaches, we see again the tendency that fewer layers for the directions (*e.g*. Additive Separate) is better than more (*e.g*. Single MLP). Our enhancement for the additive split (*enh.*) as introduced in Sec. 4.4 shows consistent improvements of the respective vanilla additive model.

References

[1] Tomas Akenine-Möller, Eric Haines, Naty Hoffman, Angelo Pesce, Michał Iwanicki, and Sébastien Hillaire. *Real-Time Rendering 4th Edition*. A K Peters/CRC Press, 2018.

[2] Brent Burley and Walt Disney Animation Studios. Physically-based shading at disney. In *International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 2012.

[3] Tomoki Ichikawa, Yoshiki Fukao, Shohei Nobuhara, and Ko Nishino. Fresnel microfacet BRDF: Unification of polari-radiometric surface-body reflection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[4] Alec Jacobson. Common 3D test models. https://github.com/alecjacobson/common-3d-test-models, 2020.

[5] Lukas Koestler, Daniel Grittner, Michael Möller, Daniel Cremers, and Zorah Lähner. Intrinsic neural fields: Learning functions on manifolds. In *European Conference on Computer Vision (ECCV)*, 2022.

[6] Eric P Lafortune and Yves D Willems. Using the modified phong reflectance model for physically based rendering. *Katholieke Universiteit Leuven. Departement Computerwetenschappen*, 1994.

[7] Guillaume Lavoué, Nicolas Bonneel, Jean-Philippe Farrugia, and Cyril Soler. Perceptual quality of BRDF approximations: dataset and metrics. *Computer Graphics Forum*, 2021.

[8] Min Li, Zhenglong Zhou, Zhe Wu, Boxin Shi, Changyu Diao, and Ping Tan. Multi-view photometric stereo: A robust solution and benchmark dataset for spatially varying isotropic materials. *IEEE Transactions on Image Processing (TIP)*, 2020.

[9] Wojciech Matusik, Hanspeter Pfister, Matt Brand, and Leonard McMillan. A data-driven reflectance model. *ACM Transactions on Graphics (TOG)*, 2003.

[10] Szymon M Rusinkiewicz. A new change of variables for efficient BRDF representation. In *Eurographics Symposium on Rendering (EGSR)*, 1998.

[11] Raif M. Rustamov. Laplace-Beltrami eigenfunctions for deformation invariant shape representation. In *Symposium on geometry processing*, 2007.

[12] Kripasindhu Sarkar, Marcel C. Bühler, Gengyan Li, Daoye Wang, Delio Vicini, Jérémy Riviere, Yinda Zhang, Sergio Orts-Escolano, Paulo F. U. Gotardo, Thabo Beeler, and Abhimitra Meka. LitNeRF: Intrinsic radiance decomposition for high-quality view synthesis and relighting of faces. In *SIGGRAPH Asia*, 2023.

[13] Christophe Schlick. An inexpensive BRDF model for physically-based rendering. In *Computer Graphics Forum*, 1994.

[14] Bruce Smith. Geometrical shadowing of a random rough surface. *IEEE Transactions on Antennas and Propagation*, 1967.

[15] Alejandro Sztrajman, Gilles Rainer, Tobias Ritschel, and Tim Weyrich. Neural BRDF representation and importance sampling. In *Computer Graphics Forum*, 2021.

[16] Kenneth E. Torrance and Ephraim M. Sparrow. Theory for off-specular reflection from roughened surfaces. *Journal of the Optical Society of America (JOSA)*, 1967.

[17] TS Trowbridge and Karl P Reitz. Average irregularity representation of a rough surface for ray reflection. *Journal of the Optical Society of America (JOSA)*, 1975.

[18] Bruce Walter, Stephen R Marschner, Hongsong Li, and Kenneth E Torrance. Microfacet models for refraction through rough surfaces. In *Eurographics Symposium on Rendering (EGSR)*, 2007.

[19] Xiuming Zhang, Pratul P. Srinivasan, Boyang Deng, Paul E. Debevec, William T. Freeman, and Jonathan T. Barron. NeRFactor: neural factorization of shape and reflectance under an unknown illumination. *ACM Transactions on Graphics (TOG)*, 2021.

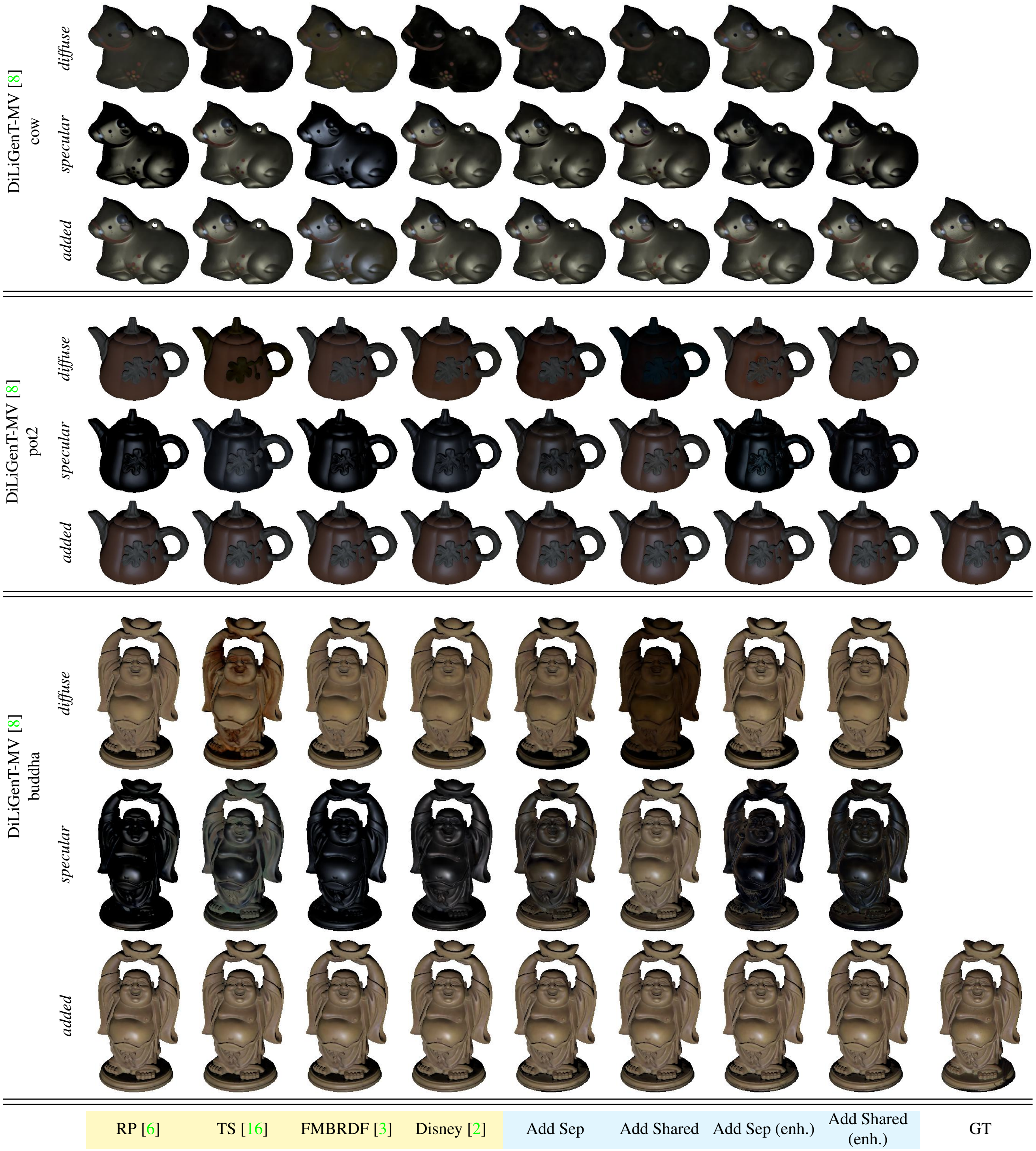


Figure C.3. Renderings of the diffuse and the specular parts separately for all additive models. Note that for the models with the enhanced additive strategy (*enh.*), the diffuse part is already weighted with ξ. Also shown are the combined rendering (*added*) and the ground truth image (*GT*). All models show a reasonable split into diffuse albedo and specular parts. For the vanilla methods of the purely neural category (Additive Separate and Additive Shared), we observe the tendency to represent more appearance in the specular part, which, as can be seen by the other models, does not seem necessary. Our enhancement for the additive split (*enh.*) as introduced in Sec. 4.4 and in particular the regularizers discussed in Sec. 4.4 and Appendix B.4 seem to improve on the disentanglement of diffuse and specular components. Note that for the cow object, a colored (RGB) specular component seems necessary to reconstruct its appearance. The FMBRDF model [3], which is based on a scalar specular term, shows an unnatural glow. Note that we observed similar behavior for the scalar specular term proposed by NeRFactor [19], as discussed in Appendix C.4.

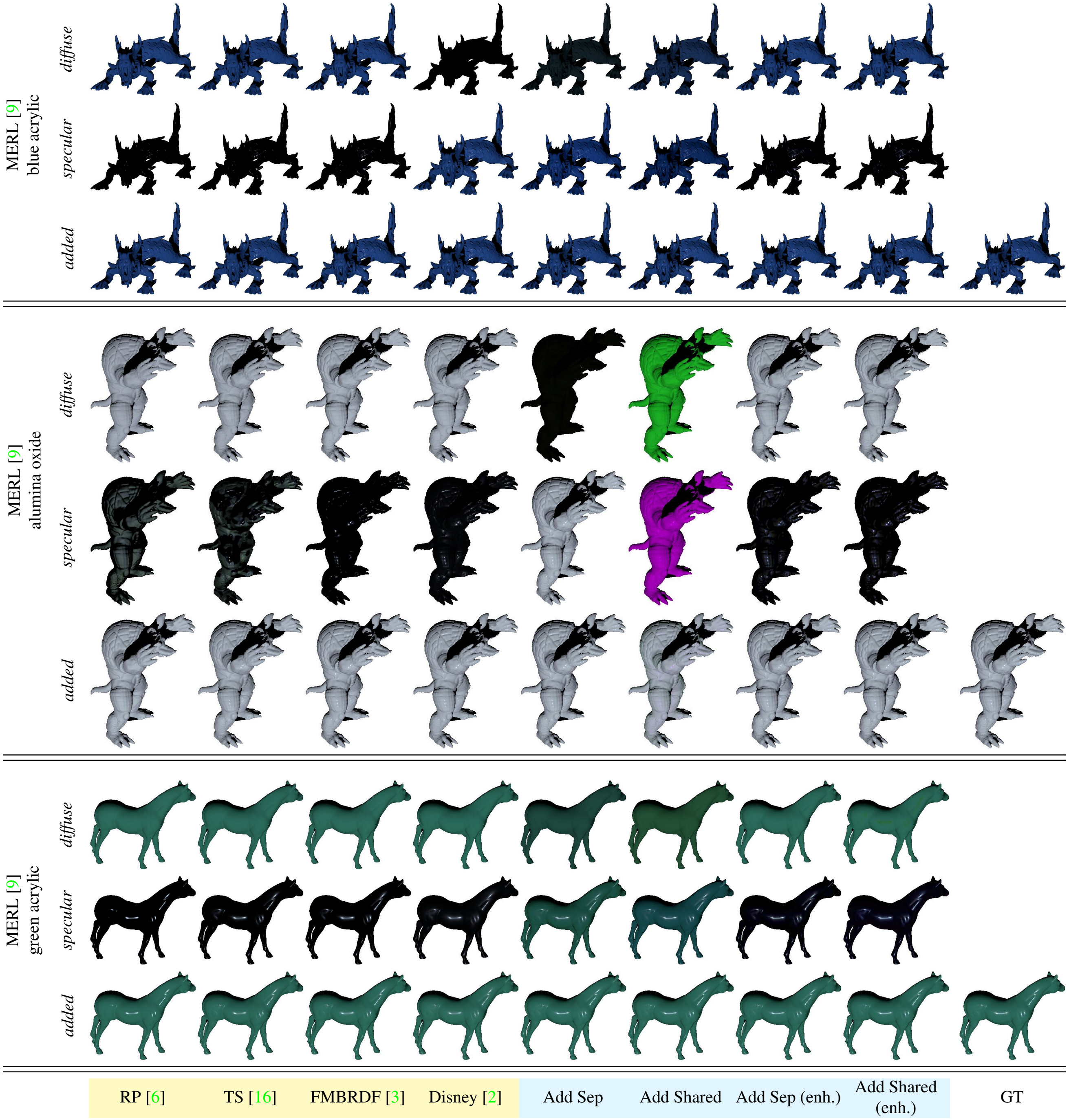


Figure C.4. Renderings of the diffuse and the specular parts separately for all additive models. Note that for the models with the enhanced additive strategy (*enh.*), the diffuse part is already weighted with ξ. Also shown are the combined rendering (*added*) and the ground truth image (*GT*). The figure shows non-metallic objects. Most models show a reasonable split into diffuse albedo and specular highlights. However, the figure reveals an issue that we observed occasionally for the vanilla purely neural models based on an additive split: For some materials, like the alumina oxide in this case, there seems to be an ambiguity that allows the model to perform an unreasonable "color-split". While the added result yields the correct colors for all of our experiments, and we do not observe a reduced reconstruction quality, this ambiguity might cause problems in specific cases. Note that our enhancement for the additive split (*enh.*) as introduced in Sec. 4.4 and in particular the regularizers discussed in Sec. 4.4 and Appendix B.4 eliminate this issue.

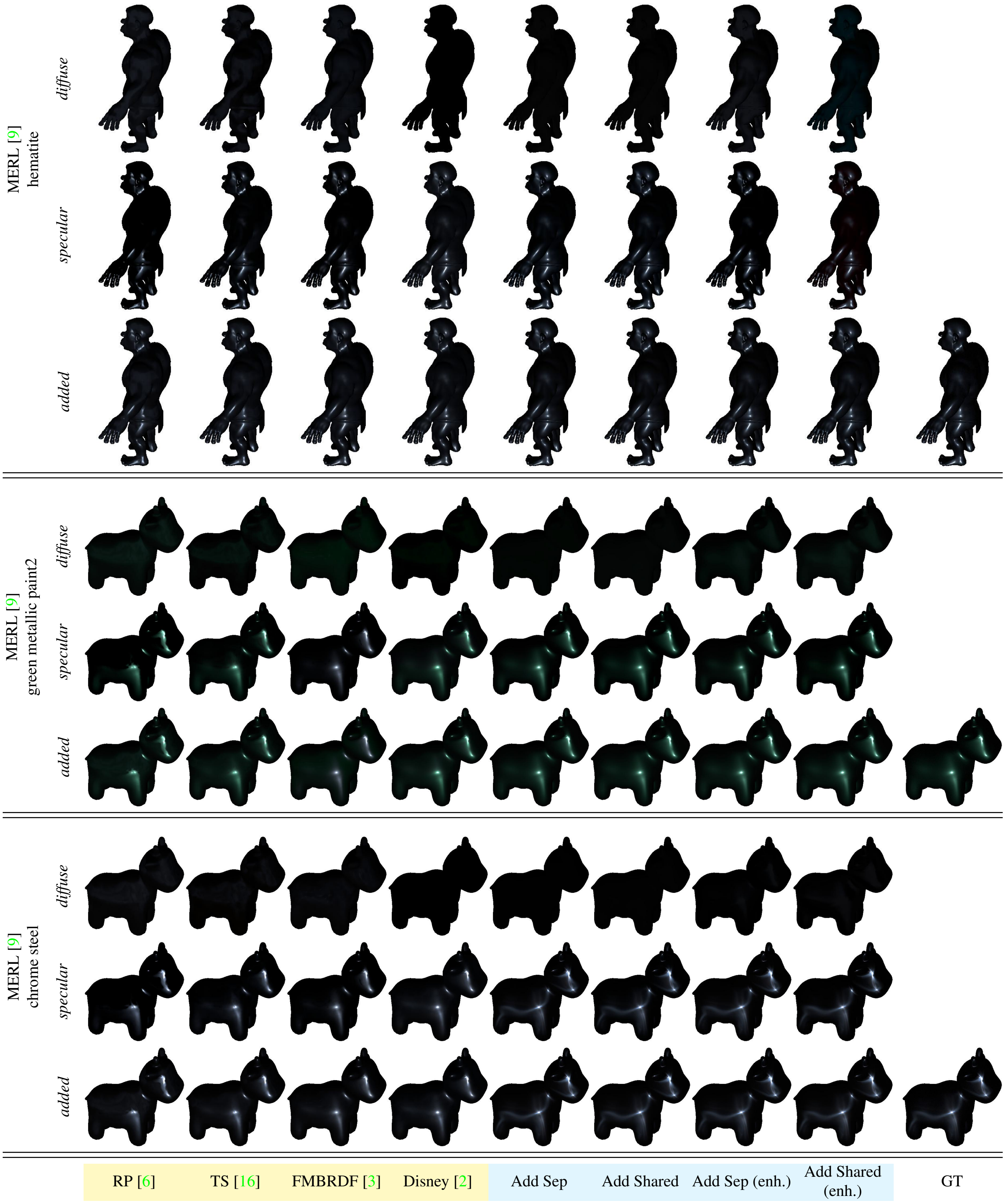


Figure C.5. Renderings of the diffuse and the specular parts separately for all additive models. Note that for the models with the enhanced additive strategy (*enh.*), the diffuse part is already weighted with ξ. Also shown are the combined rendering (*added*) and the ground truth image (*GT*). The figure shows metallic objects. This type of material shows almost no subsurface scattering, due to the free electrons [1]. All models are able to replicate this behavior, as can be seen clearly by the almost non-existent contribution of the diffuse part.

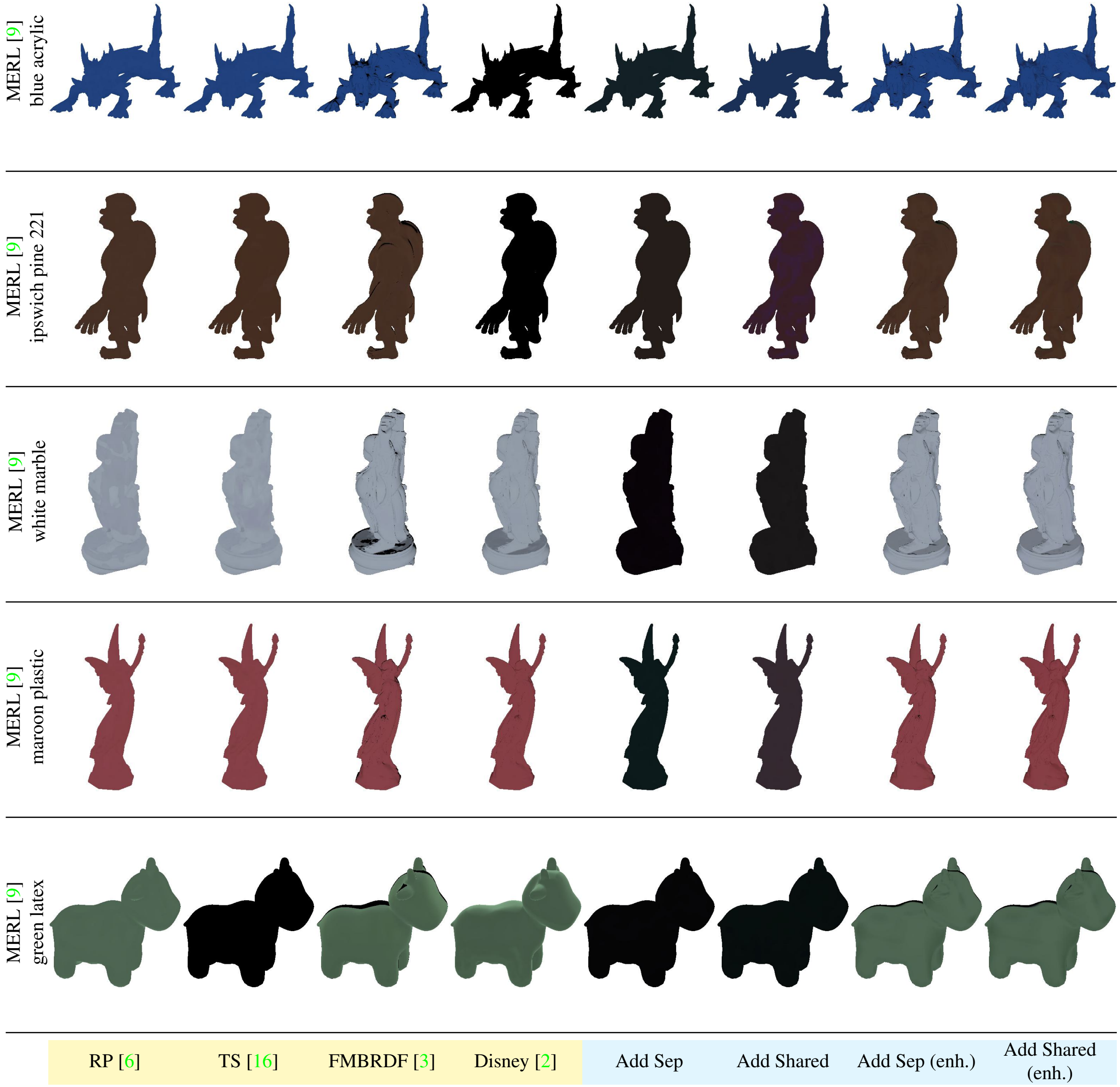


Figure C.6. Analysis of the spatial variance of the reconstructed BRDFs. Shown are the albedos rendered without the cosine term for five spatially uniform objects of the MERL-based semi-synthetic dataset. We see that overall all models are able to capture the uniformity of the BRDF well and show minimal spatial variation. The results also reveal the ambiguity introduced by an additive splitting strategy, which allows the model to capture the appearance solely by the specular term, leading to almost zero albedo. While this is particularly true for the purely neural models, we also occasionally observe it for other models (*e.g*. green latex for Torrance-Sparrow or Disney for Ipswitch pine).

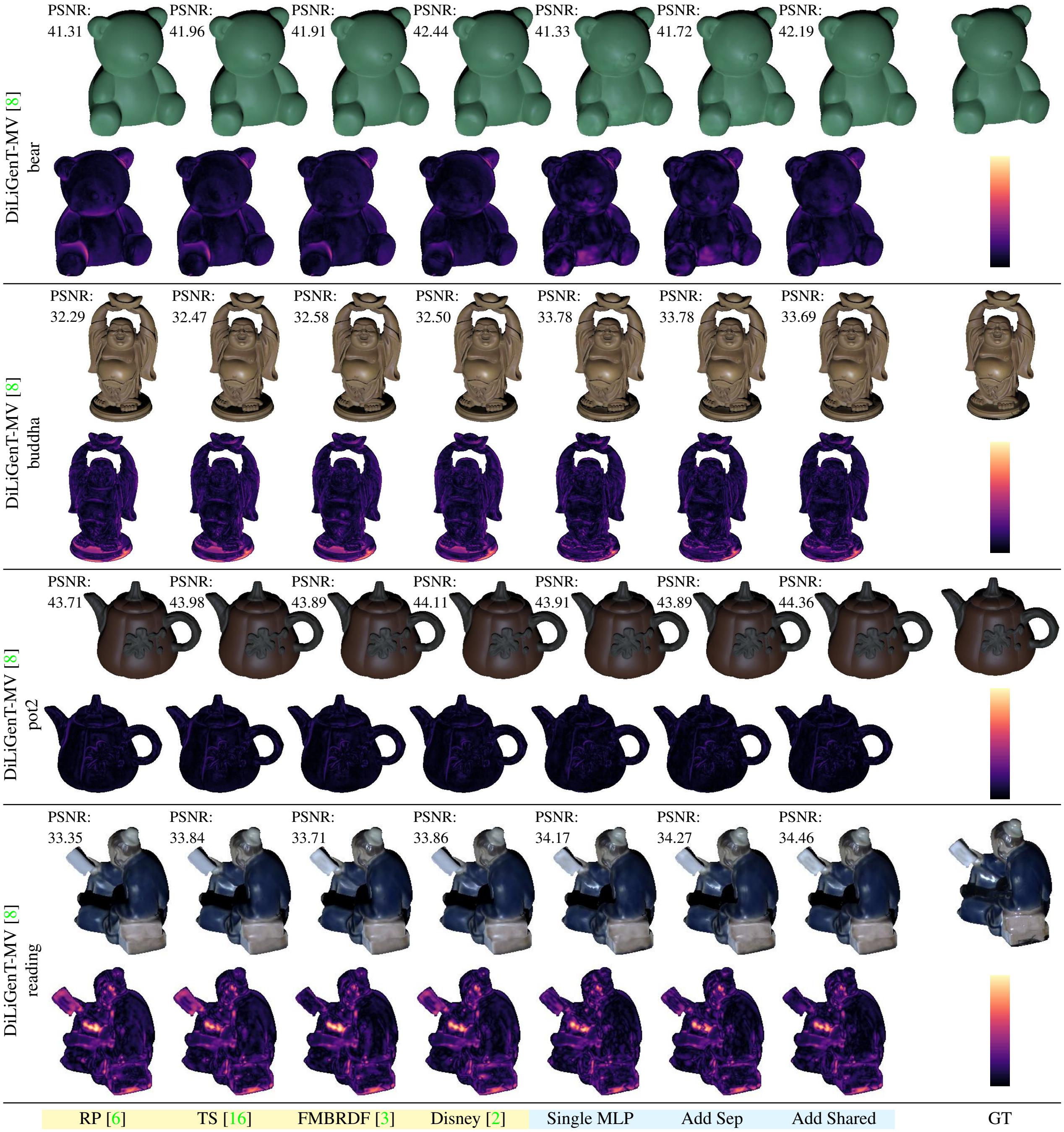


Figure C.7. Qualitative evaluation for the four remaining objects of the DiLiGenT-MV dataset [8] not presented in the main paper. Shown are renderings in sRGB space with the corresponding PSNR values and the ꟻLIP error maps for the sRGB renderings. Both, purely neural approaches (🟦) and parametric models (🟨) show errors in similar regions - recesses in particular - which makes interreflections a likely cause. We observe a tendency of overfitting for the purely neural models for the bear object, which is also visible as artifacts in the renderings. Among the purely neural models, approaches with fewer layers for the directions (*e.g*. Additive Separate) are less affected than architectures with more layers for the directions (*e.g*. Single MLP). This matches the observation in Sec. 6.2 that models with fewer layers for the directions are more robust for the potentially more noisy real-world dataset.

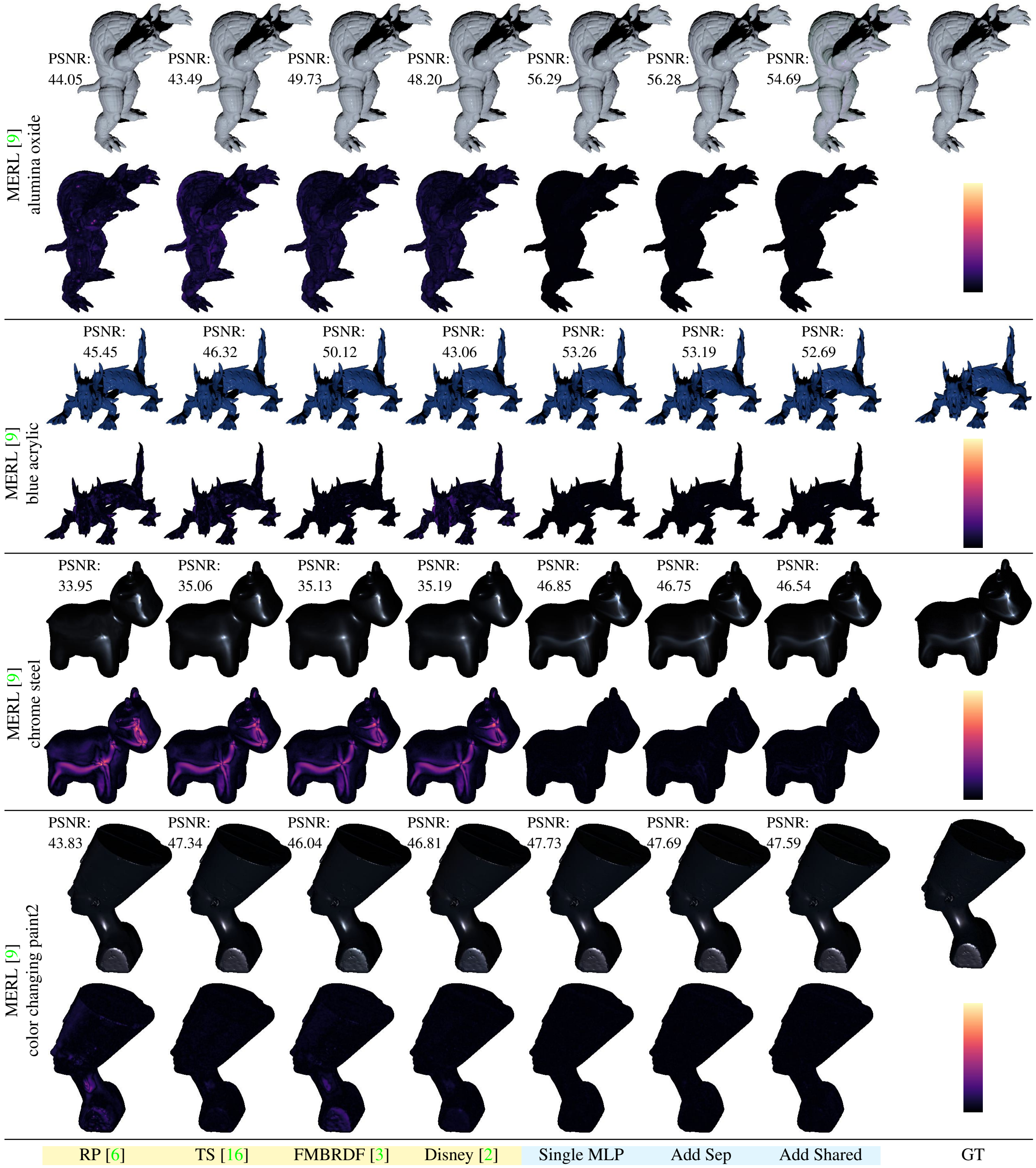


Figure C.8. Qualitative evaluation of the reconstruction for four BRDFs from the MERL database [9] uniformly rendered on common test meshes from [4]. Shown are renderings in sRGB space with the corresponding PSNR values and the ꟻLIP error maps for the sRGB renderings. Purely neural approaches (🟦) show superior results over the parametric models (🟨).

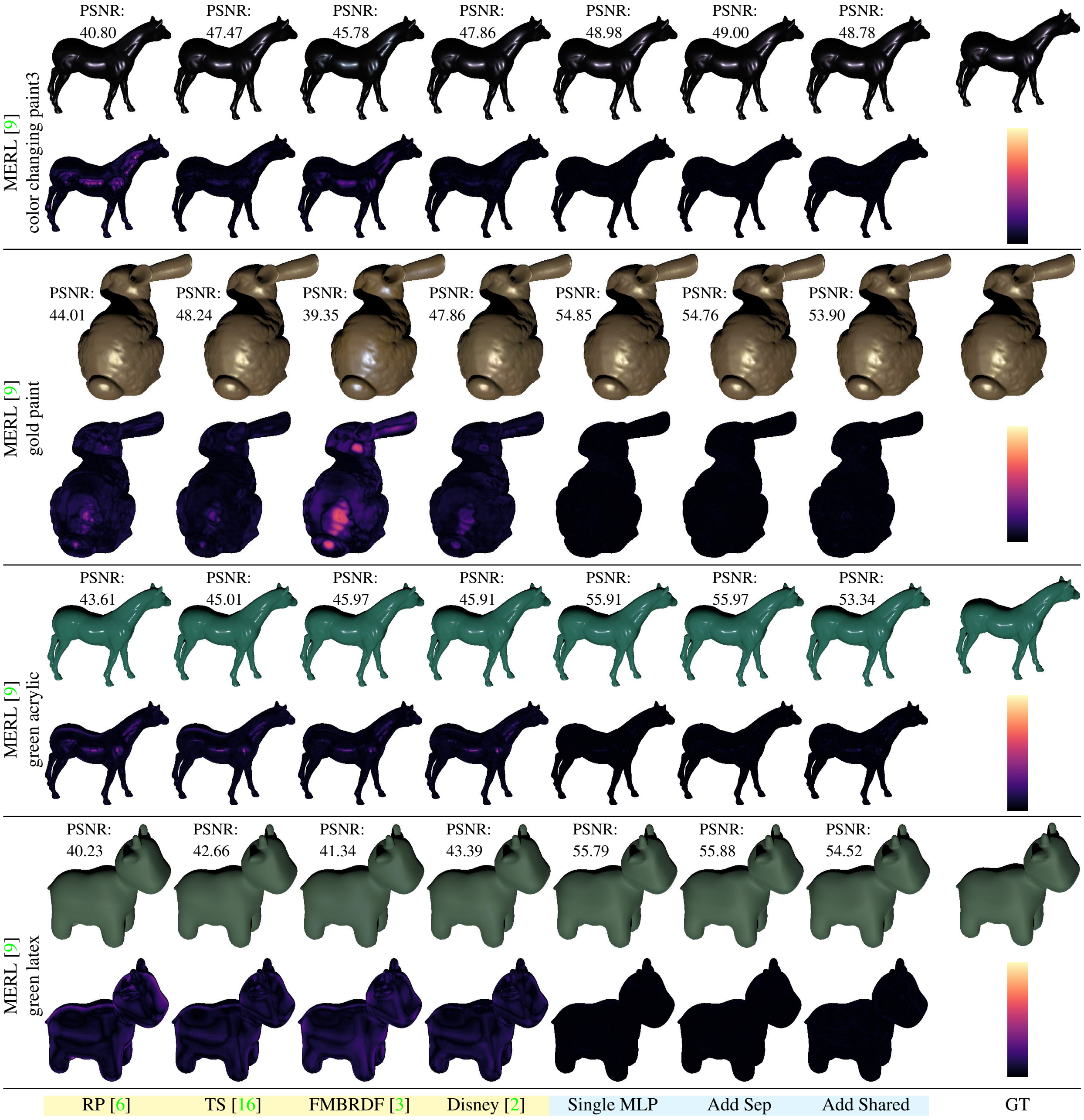


Figure C.9. Qualitative evaluation of the reconstruction for four BRDFs from the MERL database [9] uniformly rendered on common test meshes from [4]. Shown are renderings in sRGB space with the corresponding PSNR values and the ꟻLIP error maps for the sRGB renderings. Purely neural approaches (🟦) show superior results over the parametric models (🟨).

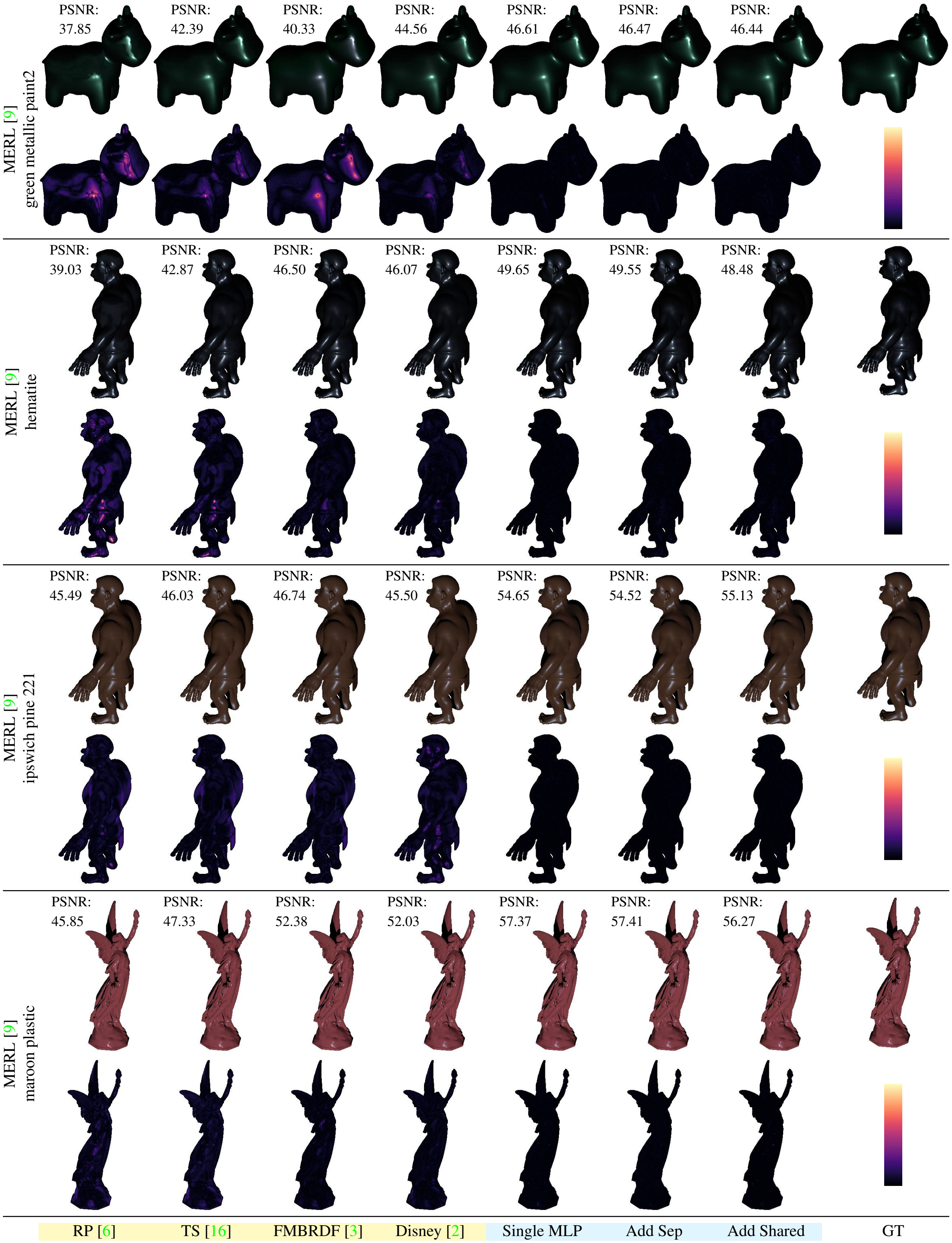


Figure C.10. Qualitative evaluation of the reconstruction for four BRDFs from the MERL database [9] uniformly rendered on common test meshes from [4]. Shown are renderings in sRGB space with the corresponding PSNR values and the ꟻLIP error maps for the sRGB renderings. Purely neural approaches (🟦) show superior results over the parametric models (🟨).

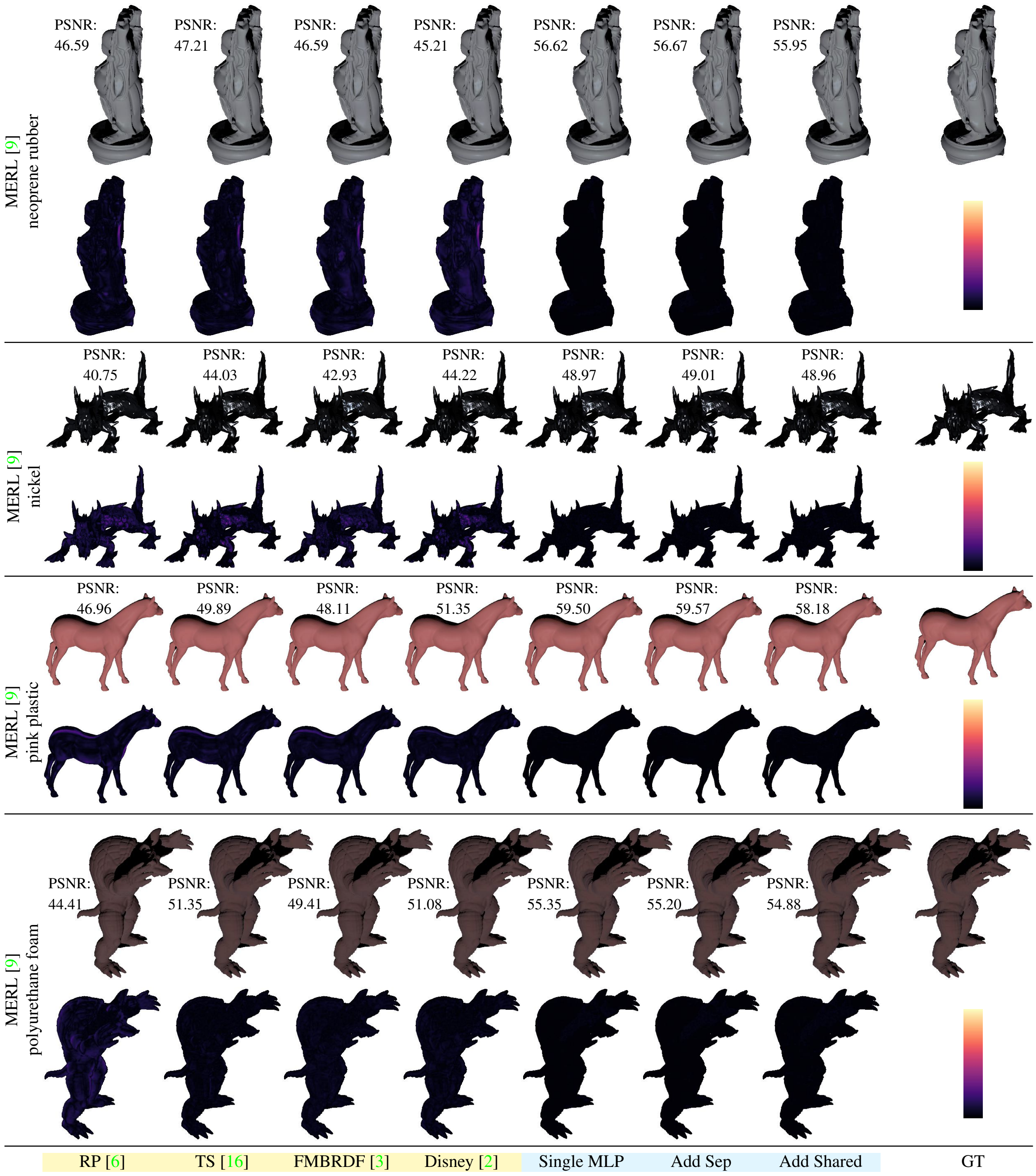


Figure C.11. Qualitative evaluation of the reconstruction for four BRDFs from the MERL database [9] uniformly rendered on common test meshes from [4]. Shown are renderings in sRGB space with the corresponding PSNR values and the ꟻLIP error maps for the sRGB renderings. Purely neural approaches (🟦) show superior results over the parametric models (🟨).

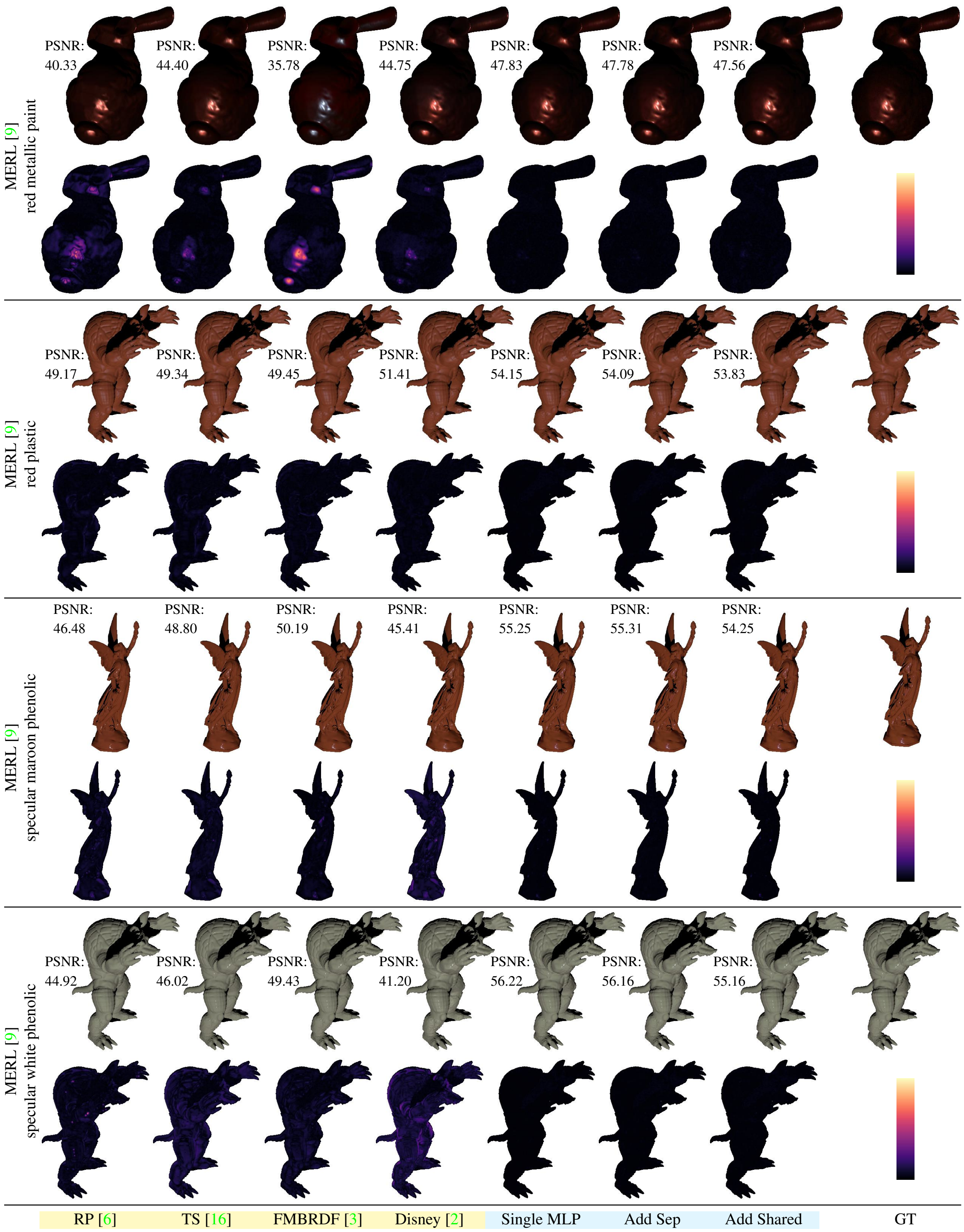


Figure C.12. Qualitative evaluation of the reconstruction for four BRDFs from the MERL database [9] uniformly rendered on common test meshes from [4]. Shown are renderings in sRGB space with the corresponding PSNR values and the ꟻLIP error maps for the sRGB renderings. Purely neural approaches (🟦) show superior results over the parametric models (🟨).

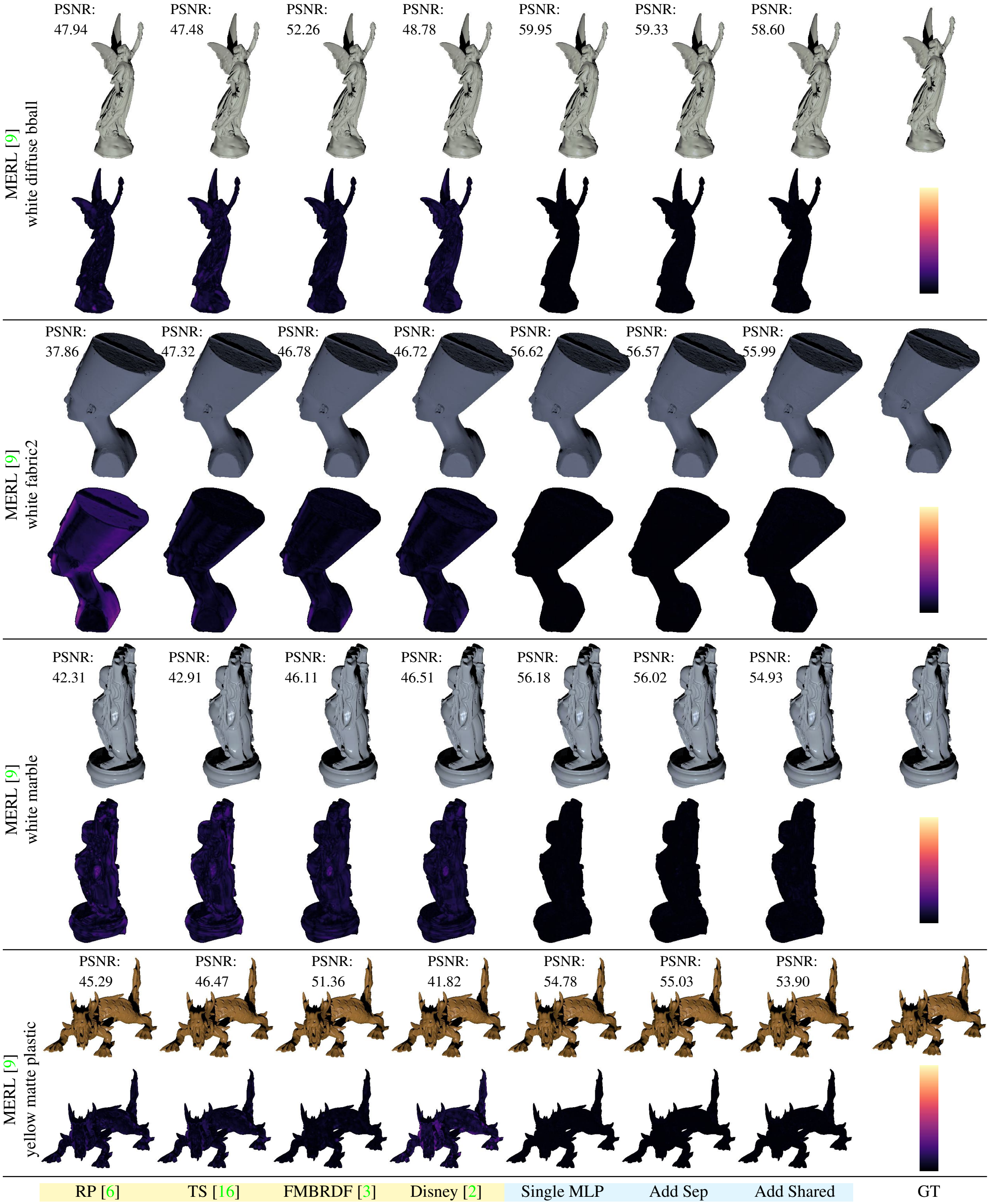


Figure C.13. Qualitative evaluation of the reconstruction for four BRDFs from the MERL database [9] uniformly rendered on common test meshes from [4]. Shown are renderings in sRGB space with the corresponding PSNR values and the ꟻLIP error maps for the sRGB renderings. Purely neural approaches (🟦) show superior results over the parametric models (🟨).