Supplementary Material
PANDORA: Polarization-Aided Neural Decomposition Of Radiance

Akshat Dave[1], Yongyi Zhao[1], and Ashok Veeraraghavan[1]

Rice University, Houston TX 77005, USA
{ad74, yongyi, vashok}@rice.edu

1 Forward Model Derivation

In this section, we elaborate on the derivation of exitant Stokes vector as a function of diffuse and specular radiance as described in Eq. 3 of the main manuscript.

Diffuse Component In Eq. 2, we decompose the outgoing Stokes vector into diffuse and specular components. First we focus on the diffuse component. From the definition of H_d for pBRDF model [4] and the illumination Stokes vector defined in eq. 1, we obtain

$$H_d \cdot S_i = \rho(\mathbf{n} \cdot \mathbf{i}) L_i T_i^+ T_i^- \begin{bmatrix} T_o^+ \\ T_o^- \alpha_o \\ -T_o^- \delta_o \\ 0 \end{bmatrix} , \tag{1}$$

where ρ is the diffuse albedo, $\mathbf{n}$ is the surface normal and $\mathbf{i}$ is the incident illumination direction. With ϕ_n denoting the exitant azimuth angle w.r.t. the surface normal, we define α_o and δ_o as

$$\begin{aligned} \alpha_o &= \cos\left(2\phi_{\mathrm{n}}\right) \\ \delta_o &= \sin\left(2\phi_{\mathrm{n}}\right) \end{aligned} \tag{2}$$

We denote the term $\rho(\mathbf{n} \cdot \mathbf{i}) L_i T_i^+ T^+$ as the diffuse intensity L_D. The term $H_d \cdot S_i$ is independent of the viewing direction. Thus we obtain the first component of Eq.3

$$\int_{\Omega} H_d \cdot S_i(\mathbf{x}, \omega_i) d\omega = L_d \begin{bmatrix} 1 \\ T_o^- / T_o^+ \cos(2\phi_n) \\ -T_o^- / T_o^+ \sin(2\phi_n) \end{bmatrix} \tag{3}$$

Specular Component The specular exitant Stokes vector is obtained by substitution of H_s as defined in the pBRDF model [4] and S_i from eq. 1.

$$H_s \cdot S_i = L_i \frac{k_s D G}{4(\mathbf{n} \cdot \mathbf{o})} \begin{bmatrix} R^+ \\ R^- \chi_o \\ R^- \gamma_o \end{bmatrix} . \tag{4}$$

where k_s is the specular coefficient, $\mathbf{o}$ is the exitant direction, D is the microfacet distribution and G is the microfacet shadowing term. With φ_h and φ_h denoting the incident and exitant azimuth angle w.r.t. the half angle $\mathbf{h}$ respectively, we define χ_o and γ_o as

$$\begin{aligned} \chi_h &= \sin(2\varphi_h) \\ \gamma_h &= \cos(2\varphi_h) \end{aligned} \tag{5}$$

We denote $f_s = \dfrac{k_s DGR+}{4(\mathbf{n}\cdot\mathbf{o})}$. Theoretically χ and γ, depend on the half angles and not the geometric surface normals of the object. In practice, we observe that for realistic values of the roughness, χ and γ do not significantly deviate from the value obtained using surface normals instead of the half angle,i.e. $\chi_h \approx \sin(2\phi_h)$ and $\gamma_h \approx \cos(2\phi_n)$. As a resultm

$$\int (H_s \cdot S_i) di = L_i \frac{k_s DG}{4(\mathbf{n}\cdot\mathbf{o})} \begin{bmatrix} R^+ \\ R^-\chi_o \\ R^-\gamma_o \end{bmatrix} \int f_s L_i di. \tag{6}$$

We denote $R^+ \int f_s L_i di$ as the specular radiance L_s and obtain the specular component of the output Stokes vector

2 Implementation Details

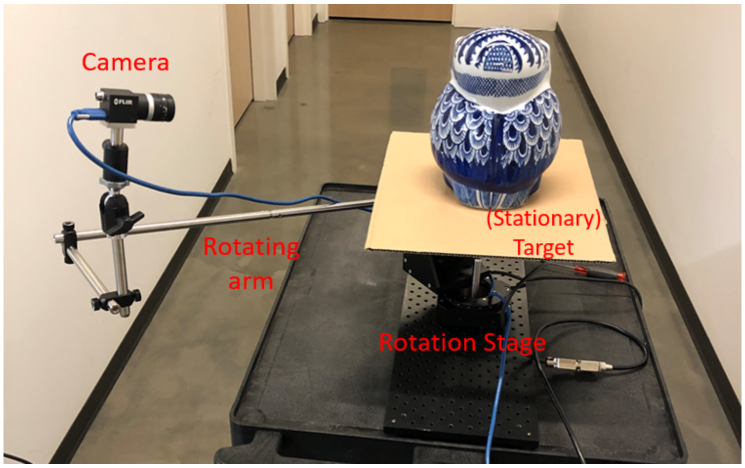


Fig. 1: **Experimental Setup:** Above, is an image of our experimental setup. The target object is placed on the stationary section of a rotation stage, which is attached to an extended arm and the snapshot polarimetric camera. The camera capture polarimetric images from multiple angles under unstructured lighting while the target object remains still.

Real world data was captured with a Blackfly S USB3 camera with Sony IMX250MYR Polarization-RGB sensor [1]. 35 images were captured for the Ball-Cup, Owl and Gnome objects under different lighting conditions as described in

Table 2. The camera was placed along multiple angles distributed roughly equally along a circle around the target object using a portable setup as shown in Fig. 1. To capture the ground truth illumination map as shown in Fig. 4 last row, we use the same setup and flip the camera so that it points outside instead of the scene. Fish eye lens is used to increase the field of view and multi-view images are captured and stitched together to obtain the ground truth illumination map.

Rendering data generation Simulated data is generated using the Mitsuba2 renderer [7]. In Mitsuba2, we are able to set the material properties, camera angles, illumination, and imaging modality (polarized or unpolarized). We use a brdf that possesses equally weighted diffuse and dielectric (specular) components. We use 45 camera views distributed over all azimuth angles, and range from 25 to 50 degrees in elevation. Our two ground truth targets were a standard sphere and a bust shape obtained from [8]. The camera views are shown in Fig. 2.

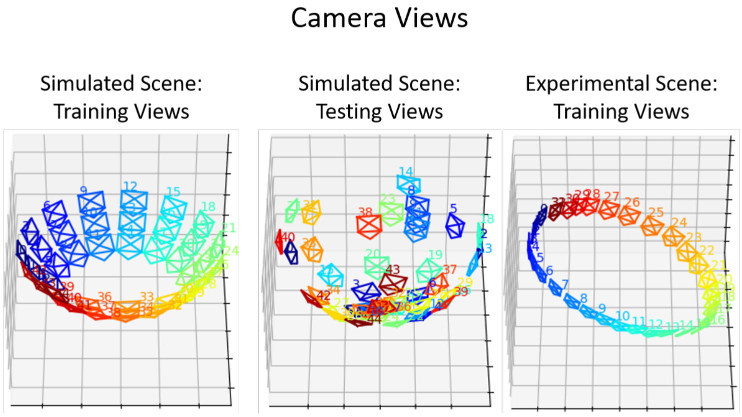


Fig. 2: **Camera Views** Above, we show the camera positions for both the simulated and experimentally captured data.

Training details All training and testing was conducted on a server containing Nvidia 2080 Ti's. As stated in our main body, our DiffNet, MaskNet, RoughNet, and IllumNet were standard MLPs with 4 layers and a width of 512. Our SDFNet was an 8-layer MLP with a width of 256 and a single skip connection in the 4-th layer. For the Stokes loss, $\mathcal{L}_{\text{stokes}}$, we chose L1 norm instead of L2 norm as the captured images have sharp intensity variations and L2 norm could result in smoothing of these features [2].Our training procedure uses several hyperparameters. The most relevant parameters include the weightage of the stokes vector loss, the weight of the mask network loss, the number of warm up iterations (before the stokes vector and specular components are estimated), and the total number of iterations. For real-world data we use 1000 warm-up iterations and 100,000 total iterations, while for simulated data we use 1500 warm-up iterations and 50,000 total iterations. We empirically found that a mask loss weightage and stokes loss weightage, w_s of 1.0 and 0.1, respectively, produced high-quality results. We observe a trade-off in deciding the optimal value of w_s. If w_s is too small, the polarimetric cues are neglected resulting in artefacts along specular highlights. If w_s is too high, the Stokes components s_2 and s_3 get higher weightage than s_0. s_2 and s_3 can be noisy for unpolarized regions resulting in noisy reconstructions for large values of w_s. The diffuse and mask networks used a sigmoid activation function, while the specular and roughness networks used a softplus activation function to avoid vanishing gradients. Finally, for our SDFNet, MaskNet, RoughNet, and IllumNet, we used the frequency embeddings described by Mildenhall et al [6]. The frequencies of the embeddings were sampled in log-space from $2^0 - 2^6$ for the SDFNet and from $2^0 - 2^{10}$ for the MaskNet, RoughNet, and DiffNet. The integrated directional embeddings were used to embed the directional coordinates for the IllumNet, as described in more detail in the subsequent section.

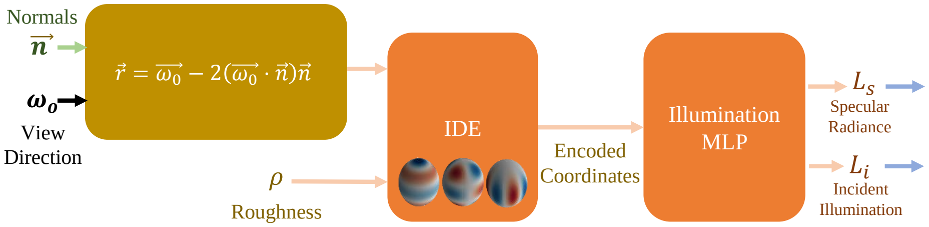


Fig. 3: **Illumination Network Design**: The illumination network accepts the reflected direction vector and the predicted surface roughness as input. The reflected direction is calculated from the surface normal and viewing direction as shown above. The roughness and direction vector are encoded by the IDE before it is passed to the MLP which generates the predicted illumination and radiance based on fresnel reflectance.

Illumination Network Design The illumination network is responsible for calculating the incident illumination (the environment map) and the specular radiance, which is derived from Fresnel reflectance. To do this, the network accepts the reflected direction and the roughness as input. The roughness parameter is estimated by a separate network, while the reflected direction can be calculated from the predicted surface normals (using the geometry network) and the input viewing direction. Both inputs must be encoded through the IDE to help estimate the high frequency information and incorporate the effects of the roughness parameter, i.e. increase the blurring of the predicted illumination as the roughness gets larger. For the input to our IllumNet, we used degree $L \in \{1, 2, 4\}$ spherical harmonics with order $m \in [-L, L]$ for the IDE's.

3 Additional Results

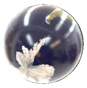
Scene	Approach	Diffuse		Specular		Mixed		Normals
		PSNR ↑ (dB)	SSIM ↑	PSNR ↑ (dB)	SSIM ↑	PSNR ↑ (dB)	SSIM ↑	MAE ↓ (°)
Bust	NeuralPIL	23.90	0.87	18.04	0.87	26.71	0.87	15.36
	PhySG	22.64	0.94	23.00	0.94	19.94	0.72	9.81
	Ours no pol no Illum	28.29	0.968	21.13	0.906	22.29	0.951	7.89
	Ours no pol	25.78	0.956	18.23	0.856	22.50	0.927	4.83
	Ours	29.53	0.973	23.63	0.912	25.97	0.951	1.95
Sphere 	NeuralPIL	13.09	0.55	12.92	0.55	20.04	0.66	38.73
	PhySG	21.76	0.76	18.90	0.76	17.93	0.70	8.42
	Ours no pol no Illum	20.65	0.76	16.23	0.76	17.11	0.72	1.91
	Ours no pol	22.20	0.83	21.30	0.87	20.87	0.82	1.92
	Ours	24.29	0.84	21.29	0.88	21.29	0.83	1.04

Table 1: **Quantiative evaluation on rendered scenes** We evaluate PANDORA with state-of-the-art and ablation methods on held-out testsets of 45 images for two rendered scenes. We report the peak average signal-to-noise ratio (PSNR) and structured similarity (SSIM) of diffuse, specular and net radiance and mean angular error (MAE) of surface normals. PANDORA consistently outperforms state-of-the-art in radiance separation and geometry estimation.

In Fig. 4, we show additional qualitative comparisons with state-of-the-art inverse rendering technique, PhySG [10], and ablation model run on intensity-only images. In Fig. 5, we highlight the advantages of PANDORA over existing mesh optimization-based polarimetric inverse rendering technique, PMVIR [11]. We also report additional quantitative metrics on simulated and real data in Table 1 and Table 2 respectively. Please refer to the supplementary html file for videos showcasing our multi-view renderings.

4 Analysis

Performance on out-of-distribution views As expected, for regions outside of the views in our training images, the estimation performs poorly. We see in Fig. 6 the network extrapolates a blob above the statue, in regions that are not heavily sampled during training. This affects our rendering when we sample rays in these regions (Fig. 6 panel 4). Finally, we see that by sampling rays only within a narrower region of interest, corresponding to locations with more training views, we obtain a correct estimate. We should note that in our main paper, the reported metrics do not account for this poor extrapolation as the images were rendered over a wider region of interest. So, the metrics were affected by artefacts in some of the rendered images shown in Fig. 6 panel 4. Metrics reported in Table 1 are with images rendered over smaller region of interest and do not have this artefact.

Effect of roughness on illumination estimation Above, we show the effects of the surface roughness on the estimated illumination map. As the surface roughness (α) increases, the associated, estimated environment map is increasingly

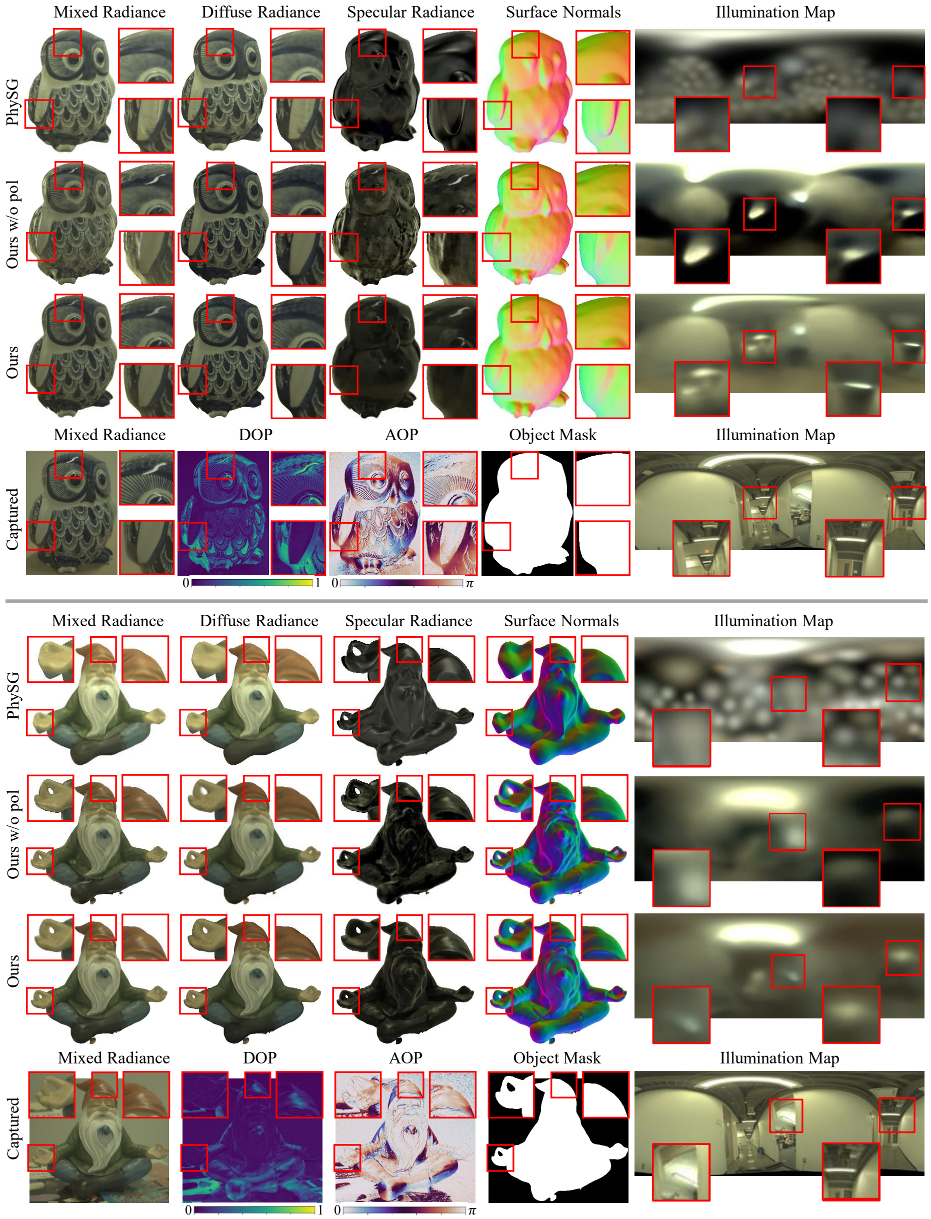


Fig. 4: **Reflectance separation,surface normal reconstruction and illumination estimatoin on real dataset** PANDORA captures high frequency details in the surface normals and accurately models the specular highlights. Please view the supplementary html file for multi-view renderings of the same.

Scene	Lighting	PhySG		Ours w\o pol		Ours	
		PSNR ↑ (dB)	SSIM ↑	PSNR ↑ (dB)	SSIM ↑	PSNR ↑ (dB)	SSIM ↑
Owl	Hallway	27.68	0.953	27.67	0.940	30.37	0.960
Gnome	Hallway	30.31	0.986	28.42	0.984	29.15	0.984
Ball-cup	Hallway	19.46	0.920	27.99	0.980	28.12	0.981
Ball-cup	Lab	14.00	0.950	23.52	0.953	26.92	0.970

Table 2: **Quantiative evaluation on real scenes** We report the average PSNR and SSIM of the rendered intensity image over the training set for objects with different material properties and under different lighting conditions. PANDORA consistently outperforms PhySG and the ablation model that is devoid of the polarimetric cues.

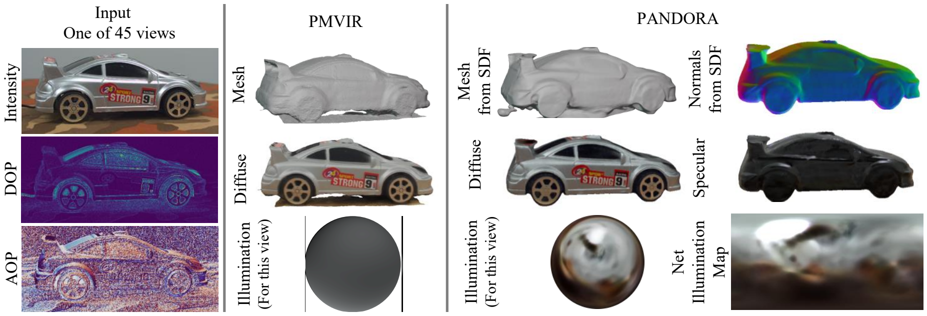


Fig. 5: **Comparison with prior mesh-based polarimetric inverse rendering on real data** Utilizing similar multi-view snapshot polarimetric data as ours, PMVIR [11] recovers 3D mesh, diffuse color for mesh vertex and lighting based on diffuse shading. Neural implicit representations enable PANDORA to extract more from the same captured data. PANDORA learns the continous signed distance field from which mesh and surface normals can be extracted. Apart from the diffuse color, PANDORA also outputs the specular radiance. Illumination estimated from PANDORA features sharp light source and the orange floor that better explain the captured data.

blurred. The inset images show the ground truth specular reflection for each of the estimated environment maps. On the right-hand side, we show the associated spherical harmonic bases, which are used for the integrated directional encoding (IDE) [1] [9]. Recall that the IDE is used to encode the directional coordinates, which are passed as input to the illumination MLP. Increasing roughness decreases the impact of the higher frequency spherical harmonic bases, as shown on the right. This helps to supervise the desired blurring effect because the high-frequency components reduced.

---

[1] The IDE visualization was generated using the ReF-NeRF implementation, with help in implementing the spherical harmonics transform from [3, 5]

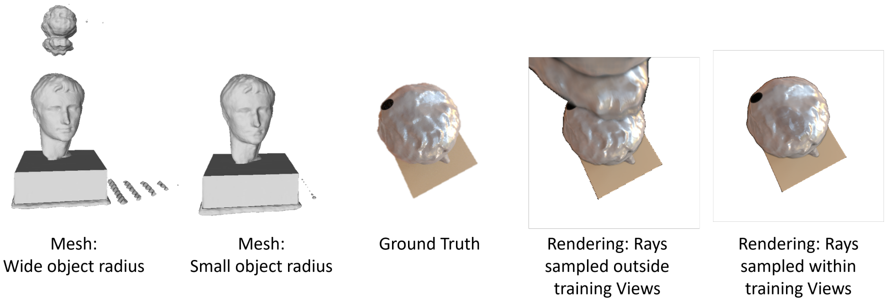

Mesh:
Wide object radius
Mesh:
Small object radius
Ground Truth
Rendering: Rays sampled outside training Views
Rendering: Rays sampled within training Views

Fig. 6: **Extrapolated Views Result:** We show the estimated mesh corresponding to regions that had lower sampled (panel 1). and higher sampled (panel 2) views. In addition, we show the resulting renderings when using more extrapolated rays (panel 4) versus without the extrapolated rays (panel 5).

Effect of roughness on polarimetric cues. In Fig. 8, we show using renderings from Mitsuba that the variation of polarimetric cues on varying roughness is less and the polarization of specular component is always distinct from the diffuse polarization under different levels of roughness.

5 Limitations

There are two main limitations to our current approach. Firstly, our method does not handle self-occlusions. This is more prominent in our simulated bust target, since the target geometry is not fully convex. We see dark patches in the estimated illumination map where the network cannot correctly estimate the illumination due to self-occlusions. In future work, this limitation may be resolved using a similar method as Verbin et al [9], in which a learnable "bottleneck" vector is used to model the target features that are not explained by other parts of the network.

Secondly, our method is that it is not able to perform re-lighting. While PANDORA is able to perform diffuse-specular radiance separation, the incident illumination is baked into these radiances and it is chalenging to estimate physically-based material properties, more specifically the material roughness and the diffuse albedo. While our network architecture possesses an α parameter that tunes the roughness appearance and models the effect of increasing roughness, such as blurred illumination map (Fig. 7), it does not truly estimate the physics-based roughness parameter.

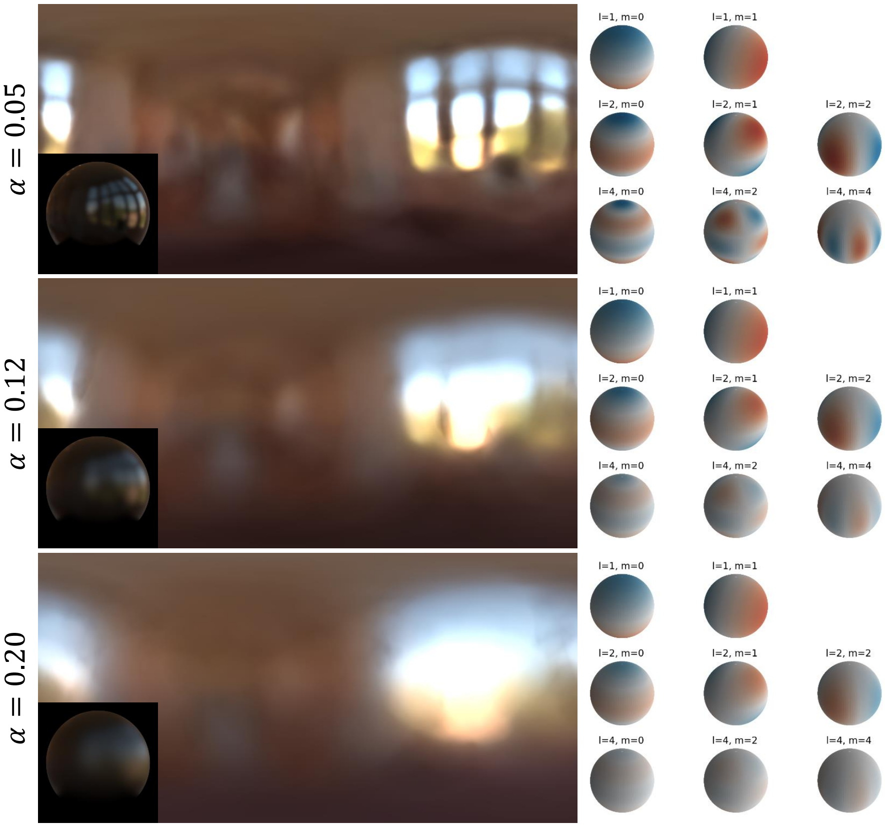


Fig. 7: **Effect of roughness on illumination estimation**: Our illumination estimation accounts for the effects of surface roughness. As the roughness (parameterized by α) increases, there is an increasing blurring effect on the estimated environment map. The inset images shows the corresponding ground truth specular reflection as the surface roughness increases. The right side shows the effect of the increasing roughness on the spherical harmonic IDE's.

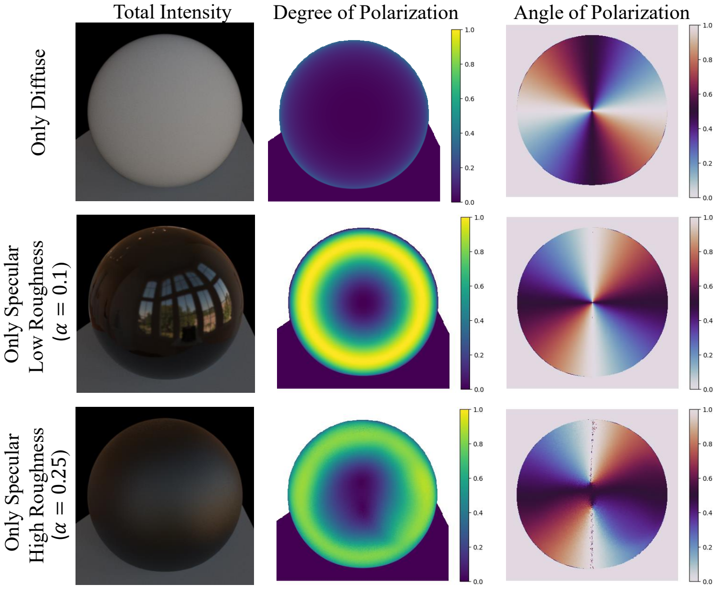


Fig. 8: **Effect of specular roughness on polarimetric cues** We render polarimetric cues for a sphere object using the pBRDF model in Mitsuba2 with varying material properties. The variation of polarimetric cues is less under the realistic range of roughness. Our insight that the specular polarization is orthogonal in angle and higher in degree than the diffuse polarization remains applicable on varying specular roughness to realistic values. *alpha* denotes the roughness parameter of the Beckmann microfacet distribution.

References

1. Sony polarization image sensors. https://www.sony-semicon.co.jp/e/products/IS/industry/product/polarization.html (2021), accessed: 2021-09-25
2. et al., H.Z.: Loss functions for image restoration with neural networks. IEEE TCI **3**(1), 47–57 (2016)
3. Alex Yu and Sara Fridovich-Keil, Tancik, M., Chen, Q., Recht, B., Kanazawa, A.: Plenoxels: Radiance fields without neural networks (2021)
4. Baek, S.H., Jeon, D.S., Tong, X., Kim, M.H.: Simultaneous acquisition of polarimetric svbrdf and normals. ACM Trans. Graph. **37**(6), 268–1 (2018)
5. Lucidrains: Se3 transformer - pytorch. https://github.com/lucidrains/se3-transformer-pytorch (2021)
6. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: ECCV (2020)
7. Nimier-David, M., Vicini, D., Zeltner, T., Jakob, W.: Mitsuba 2: A retargetable forward and inverse renderer. Transactions on Graphics (Proceedings of SIGGRAPH Asia) **38**(6) (Dec 2019). https://doi.org/10.1145/3355089.3356498
8. turbosquid: Head_augustus 3d model (2018), accessed: 2022-01-23, generated by Cone of Vision. https://www.turbosquid.com/3d-models/head-augustus-3d-model-1327693
9. Verbin, D., Hedman, P., Mildenhall, B., Zickler, T., Barron, J.T., Srinivasan, P.P.: Ref-nerf: Structured view-dependent appearance for neural radiance fields. arXiv preprint arXiv:2112.03907 (2021)
10. Zhang, K., Luan, F., Wang, Q., Bala, K., Snavely, N.: Physg: Inverse rendering with spherical gaussians for physics-based material editing and relighting. In: The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021)
11. Zhao, J., Monno, Y., Okutomi, M.: Polarimetric multi-view inverse rendering (2020)