Poppy: Polarization-based Plug-and-Play Guidance for Enhancing Monocular Normal Estimation

Irene Kim[1], Sai Tanmay Reddy Chakkera[1], Alexandros Graikos[1], Dimitris Samaras[1], and Akshat Dave[1]

Stony Brook University
{irnkim,schakkera,agraikos,samaras,dave}@cs.stonybrook.edu

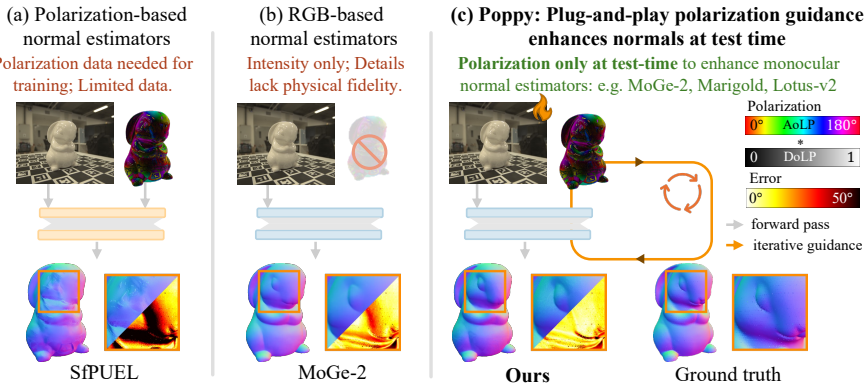


Fig. 1: Test-time polarization guidance to enhance normal estimation. (a) Polarization-based feed-forward models have limited generalizability due to the scarcity of polarization–normal training pairs. (b) RGB-only monocular normal estimators produce oversmoothed or hallucinated details on challenging surfaces – normals of the textureless bunny object appear flatter than ground truth. (c) Poppy introduces polarization guidance into pretrained RGB-only models at test time – improving normal accuracy without retraining.

Abstract. Monocular surface normal estimators trained on large-scale RGB-normal data often perform poorly in the edge cases of reflective, textureless, and dark surfaces. Polarization encodes surface orientation independently of texture and albedo, offering a physics-based complement for these cases. Existing polarization methods, however, require multi-view capture or specialized training data, limiting generalization. We introduce Poppy, a training-free framework that refines normals from any frozen RGB backbone using single-shot polarization measurements at test time. Keeping backbone weights frozen, Poppy optimizes per-pixel offsets to the input RGB and output normal along with a learned

reflectance decomposition. A differentiable rendering layer converts the refined normals into polarization predictions and penalizes mismatches with the observed signal. Across seven benchmarks and three backbone architectures (diffusion, flow, and feed-forward), Poppy reduces mean angular error by 23–26% on synthetic data and 6–16% on real data. These results show that guiding learned RGB-based normal estimators with polarization cues at test time refines normals on challenging surfaces without retraining.

Keywords: surface normal estimation · polarization imaging · shape from polarization · test-time guidance · physics-based guidance

1 Introduction

Surface normal estimation is a core problem in computer vision, with applications in robotics, augmented reality, scene understanding, and 3D reconstruction. Monocular normal estimation aims to recover per-pixel 3D surface orientation from a single RGB image – an inherently ill-posed task because many surface normals can produce the same 2D appearance. Modern learning-based normal estimators reduce this ambiguity by learning geometric priors from large-scale paired RGB and normal datasets [6, 26, 34, 52, 55].

However, state-of-the-art monocular estimators fail on three common surface types where RGB cues alone are unreliable: 1) highly reflective surfaces, where view-dependent highlights are misinterpreted as geometry; 2) textureless regions, which lack spatial variation and produce oversmoothed normals; and 3) dark objects, where low signal-to-noise ratio degrades predictions. RGB-normal datasets often under-represent these surfaces, compounding these errors (Fig. 1(b)). Thus, in this paper, in addition to RGB, we capture the polarization of light, only at test time, to refine normal estimation for these challenging scenarios.

Polarization provides a physically grounded complement to RGB. When unpolarized light reflects from a surface, it becomes partially polarized depending on surface orientation and material – making polarization informative for the failure cases above. Classical shape-from-polarization (SfP), however, is ill-posed: the azimuthal flip (π) ambiguity and the diffuse–specular ($\pi/2$) ambiguity together yield four candidate normals from a single measurement. Prior classical methods resolve these ambiguities by acquiring additional measurements – varying the illumination [1, 4, 17, 42], capturing multiple views [8, 15, 25, 36], or capturing additional modalities [28, 33] – increasing the capture burden beyond a single snapshot.

Learning-based SfP methods [5, 35, 39] aim to retain single-snapshot capture by leveraging data-driven priors learned from curated polarization–normal training pairs. These approaches, however, require paired polarimetric training data that is expensive to collect and narrow in scene diversity, which limits generalization to new materials and environments (Fig. 1a). Neither classical nor learned SfP methods exploit the strong geometric priors already embedded in modern RGB normal estimators.

To bridge this gap, we introduce Poppy, a training-free framework that guides any differentiable RGB normal estimator with physics-based polarization constraints at test time (Fig. 1c). The weights of the normal estimator backbone are kept frozen. Poppy instead introduces learnable parameters before and after the network so that the predicted normals align with the observed polarization signal. Poppy is plug-and-play: it applies to diffusion-based [34], flow-based [26], and feed-forward [52] backbones without retraining or architectural changes. The pretrained backbone initializes normals using global scene structure from RGB as orientation constraints, resolving the azimuthal ambiguity; polarization guidance then corrects normal estimation errors that RGB cues alone miss.

Concretely, Poppy adds a learnable per-pixel image offset to the network input and a learnable per-pixel normal offset to the network output. The image offset acts as a global perturbation: input changes propagate through the backbone, steering the predicted geometry at a scene-wide scale. The normal offset refines high-frequency detail that the backbone misses. A differentiable polarization rendering layer converts the refined normals and a learned specular radiance into Stokes vectors, a representation of the polarization state, resolving the remaining diffuse–specular ambiguity. The loss between the predicted and measured Stokes vectors is minimized by optimizing the image offset, normal offset, and specular radiance maps.

We evaluate Poppy extensively across seven benchmarks – SfPUEL [39] NeRSP [25], NeISF [36], and DeepPol [16] (synthetic); SfPUEL, NeRSP, and PISR [8] (real) – covering most publicly available polarization–normal datasets. Poppy enables consistent improvements in these benchmarks across three backbone architectures: Marigold, Lotus-v2, and MoGe-2 – with 23–26% reduction in mean angular error (MAE) for synthetic data and 6–16% reduction on real data. Poppy demonstrates improved normal quality on challenging materials, including reflective, textureless, and low albedo surfaces. The refined normals also improve downstream 3D mesh quality when used to aid an existing multi-view reconstruction method. The learned per-pixel specular radiance enables decomposition of diffuse and specular Stokes.

Our contributions are:

1. A training-free guidance framework that refines normals from RGB-based monocular estimators using polarization measurements at test time.
2. A plug-and-play guidance mechanism with learnable per-pixel image and normal offsets for global-then-local refinement of any differentiable backbone, together with a learned per-pixel specular radiance for handling diffuse–specular ambiguities.
3. Consistent improvements across three backbone architectures on seven benchmarks, demonstrating gains on reflective, textureless, and low-SNR surfaces that are typically challenging for normal estimation.

2 Related work

Monocular surface normal estimation methods predict per-pixel surface orientation from a single RGB image. Early discriminative approaches [18, 21, 43] train convolutional networks on large-scale supervision, with Omnidata [20] showing the benefit of multi-task pretraining. More recent discriminative methods incorporate stronger priors: DSINE [6] encodes per-pixel camera geometry, and MoGe [52] regresses affine-invariant point maps via a transformer backbone. Diffusion-based methods [22, 23, 34, 38] repurpose pretrained generative models for geometry estimation, with Marigold [34], Lotus [27], StableNormal [55], and GenPercept [53] among the most prominent. Despite strong average-case performance, all of these methods rely solely on RGB appearance cues and degrade on reflective, textureless, and low-albedo surfaces where photometric shading signals are ambiguous. Our work targets precisely these failure cases by introducing polarization-based physical constraints at test time.

Shape from Polarization methods recover surface normals from reflected light using diffuse and height-from-polarization models [2,46,50], specular and refractive cues [28, 33, 40, 41], or joint shading–polarization formulations [1, 4, 30, 42]. However, the polarization-to-normal mapping is inherently ambiguous due to π and $\pi/2$ symmetries in the Fresnel equations. Multi-view methods [3, 8, 13, 25, 36, 54] resolve this geometrically, while discriminative methods such as DeepSfP [5] and SfPUEL [39] learn to predict normals directly from polarization images. All of these approaches either require controlled capture, curated polarization training data, or multi-image acquisition. Most closely related to our work, PPFT [31] adapts a pretrained *depth* model to polarization inputs via LoRA fine-tuning—requiring polarization training data and modifying model weights; in contrast, Poppy targets *normals* and operates entirely at test time without retraining.

Test-time guidance steers pretrained models at inference time rather than training specialized models on new data. For diffusion models, guidance methods first imposed linear constraints by back-propagating a task-specific likelihood during the reverse diffusion, effectively steering generation towards measurement-consistent solutions [11]. More recent approaches have generalized this idea to arbitrary non-linear constraints [56], and faster, backpropagation-free sampling under constraints [24]. Guidance has been extensively used in depth-completion methods [29, 32, 49, 51], where sparse depth measurements or physics-based cues are used to guide pretrained monocular depth estimators at test time, as well as in medical image segmentation [10], where image-specific visual prompts are optimized at inference to bridge cross-domain distribution shifts without modifying model weights. Among the depth methods, Marigold-DC [51] is closest to our approach: it guides a diffusion-based depth model with sparse depth observations, whereas Poppy guides a normal estimation model with dense polarization measurements that encode shape through the Fresnel equations.

3 Background

3.1 Representing polarization

Polarization characterizes the oscillation of light waves. While natural illumination is often unpolarized, reflection from surfaces induces partial polarization that depends on surface orientation and material properties. This makes polarization a physically grounded cue for geometry estimation.

Stokes vector. The polarization state of a light ray can be represented by the Stokes vector, $\mathbf{S} = [S_0, S_1, S_2, S_3]^\top$ [12]. S_0 represents total intensity. S_1 measures the difference between horizontally and vertically polarized components. S_2 measures the difference between components polarized at 45° and 135°. S_3 represents circular polarization.

A linear polarization camera captures (I) through polarizers at multiple angles ($0°, 45°, 90°, 135°$). The linear Stokes parameters are computed as

$$S_0 = I_0 + I_{90}, \quad S_1 = I_0 - I_{90}, \quad S_2 = I_{45} - I_{135}. \tag{1}$$

In most passive imaging scenarios, circular polarization is negligible compared to linear polarization. We therefore restrict our formulation to linear polarization and use the reduced representation, $\mathbf{S} = [S_0, S_1, S_2]^\top$.

Degree and Angle of Polarization. The Degree of Linear Polarization (DoLP), denoted ρ, measures the fraction of light that is linearly polarized – ranging from 0, for unpolarized light, to 1, for fully polarized light. DoLP can be obtained from the Stokes vector as $\rho = \frac{\sqrt{S_1^2+S_2^2}}{S_0}$. The Angle of Linear Polarization (AoLP), denoted ϕ, describes the dominant polarization orientation. AoLP depends on the Stokes vector as $\phi = \frac{1}{2}\arctan(S_2, S_1)$. Together, (ρ, ϕ) provide a compact representation of the linear polarization state.

Diffuse and specular reflectance. When light reflects from a surface, it can be decomposed into diffuse and specular components. We provide an example decomposition learned from our method for visualization in Fig. 2. Diffuse reflection arises from subsurface scattering and typically produces weaker polarization, whereas specular reflection arises from surface reflection and follows Fresnel reflection laws [7]. Let L_d and L_s denote the diffuse and specular radiance components. The observed Stokes vector is modeled as the sum of diffuse and specular contributions: $\mathbf{S} = S_d + S_s$. Using the DoLP and AoLP of each component, the combined Stokes parameters can be written as [15, 37]

$$S_0 = L_d + L_s \tag{2a}$$
$$S_1 = L_d\rho_d\cos(2\phi_d) + L_s\rho_s\cos(2\phi_s) \tag{2b}$$
$$S_2 = L_d\rho_d\sin(2\phi_d) + L_s\rho_s\sin(2\phi_s)\ , \tag{2c}$$

where $(\rho_{d/s}, \phi_{d/s})$ represent diffuse/specular DoLP, AoLP respectively.

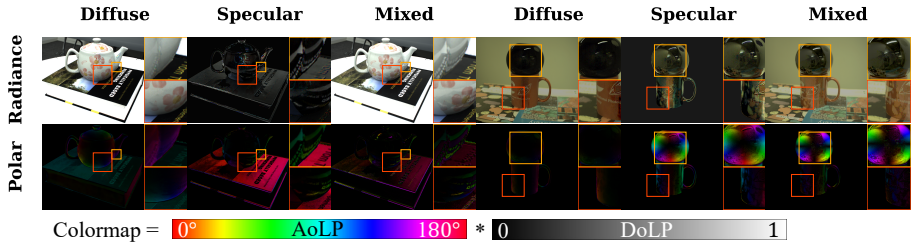


Fig. 2: Radiance decomposition. The mixed radiance S_0 is decomposed into diffuse radiance L_d and specular radiance L_s from our method. The learned L_s captures specular highlights and environment-dependent reflections (scaled 2× for clarity), while L_d retains the object's intrinsic diffuse shading and texture. From these radiance components and the predicted normals, the polarization maps (AoLP×DoLP) of the diffuse, specular, and combined components can be obtained.

3.2 Surface normals to Stokes

Polarization provides a cue for surface orientation because both DoLP and AoLP depend on the surface normal relative to the viewing direction.

Spherical coordinate conversion. Denoting the surface normal as $n = [n_x, n_y, n_z]^\top$, we convert it into spherical coordinates (θ, ψ) relative to the viewing direction v, where the elevation angle $\theta = \arccos(n \cdot v)$ and azimuth angle $\psi = \arctan(n_y, n_x)$.

DoLP from surface normals. Under Fresnel reflection theory, the degree of polarization depends on the elevation angle θ and the refractive index η (set to 1.5 per standard assumptions).

For diffuse reflection, $\rho_d(\theta) = \frac{(\eta - 1/\eta)^2 \sin^2\theta}{2+2\eta^2-(\eta+1/\eta)^2\sin^2\theta+4\cos\theta\sqrt{\eta^2-\sin^2\theta}}$, and for specular reflection, $\rho_s(\theta) = \frac{2\sin^2\theta\cos\theta\sqrt{\eta^2-\sin^2\theta}}{\eta^2-\sin^2\theta-\eta^2\sin^2\theta+2\sin^4\theta}$.

AoLP from surface normals. The angle of polarization is determined by the azimuth angle ψ. For diffuse reflections, $\phi_d = \psi$, and for specular reflections, $\phi_s = \psi + \pi/2$. The orthogonality between these components reflects the phase difference induced by Fresnel reflection.

Stokes from surface normals. Normal-to-Stokes conversion is an inverse problem of SfP. Given a surface normal n and the specular radiance L_s, we compute the diffuse and specular DoLP/AoLP $(\rho_{d/s}, \phi_{d/s})$ and diffuse radiance $L_d = S_0 - L_s$. Therefore, we can rewrite Eq. 2 in terms of n and L_s and compute Stokes vector as $\hat{\mathbf{S}} = \mathcal{F}(n, L_s)$.

Ambiguities in Shape from Polarization. Shape from Polarization suffers from intrinsic ambiguities. AoLP is invariant under a 180° rotation, leading to a π ambiguity in azimuth. Additionally, diffuse and specular reflections produce orthogonal polarization orientations, resulting in a $\pi/2$ ambiguity when the dominant

reflection component is unknown. These ambiguities motivate the integration of additional geometric priors using RGB cues, as we propose in our method.

3.3 Monocular normal estimators

A pretrained monocular estimator f predicts the surface normal as $n = f(x)$, by leveraging the rich visual priors learned from pretraining on large-scale datasets. Deep-learning models span diverse architectures: iterative diffusion-based models (*e.g.*, Marigold [34]), iterative flow-based models (*e.g.*, Lotus-v2 [26]), feed-forward transformer models (*e.g.*, MoGe-2 [52]).

4 Method

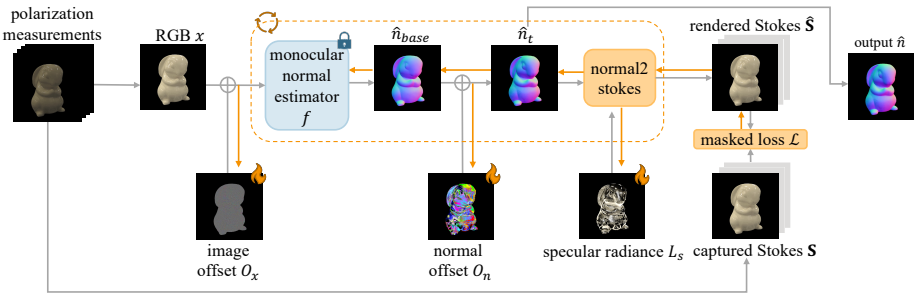


Fig. 3: Poppy pipeline. Given polarization measurements, we compute the observed Stokes map $\mathbf{S}$, extract the RGB image x, and add learnable image offset O_x to x. A frozen backbone produces base normals $\hat{n}_{\text{base}}$; a learnable normal offset O_n yields the refined estimate $\hat{n}_t = \hat{n}_{\text{base}} + O_n$. Using Fresnel equations, the predicted Stokes $\hat{\mathbf{S}}$ is computed from $\hat{n}_t$ and specular radiance L_s. We minimize the polarization consistency loss between $\hat{\mathbf{S}}$ and $\mathbf{S}$ to update the image offset O_x, normal offset O_n, and specular map L_s over T steps, while keeping backbone weights fixed.

Given polarization measurements captured by a linear polarization camera, we recover surface normals that are consistent with physical light transport while preserving the strong geometric priors learned by modern RGB-based normal estimation networks. Instead of retraining a model, we formulate normal recovery as a test-time guidance problem. During the guidance, a differentiable rendering model enforces polarization constraints.

4.1 Rendering Stokes from monocular normals

Given the input polarization measurements $I_{\{0,45,90,135\}^\circ}$, we first compute the observed Stokes vector $\mathbf{S} = [S_0, S_1, S_2]^\top$ via Eq. 1, where the unpolarized intensity S_0 serves as the RGB input x to our pipeline.

A pretrained RGB normal estimator f takes x and produces a base surface normal estimate $\hat{n}_{\text{base}} = f(x)$. To synthesize the predicted Stokes vector $\hat{\mathbf{S}}$ from a given normal $\hat{n}_{\text{base}}$, we require the specular radiance L_s, following the procedure described in Sec. 3.2. In principle, L_s can be optimized iteratively via polarization-guided refinement until the residual $|\mathbf{S}-\hat{\mathbf{S}}|$ vanishes. However, when f produces an erroneous normal estimate in geometrically challenging regions, prolonged optimization of L_s alone leads to either a persistent mismatch between the observed Stokes $\mathbf{S}$ and predicted Stokes $\hat{\mathbf{S}} = \mathcal{F}(f(x), L_s)$, or overfitting of L_s to the incorrect normal (second column of Fig. 4(a))— neither of which corrects the underlying geometric error.

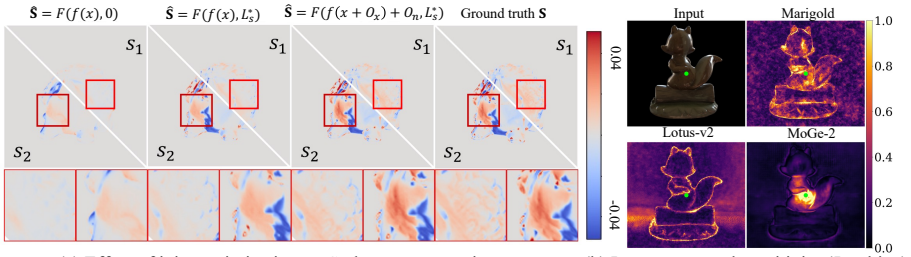
This failure mode stems from the fact that the error originates in $\hat{n}_{\text{base}}$. To address this, we seek a test-time guidance mechanism that corrects the normal estimate directly, without modifying frozen model weights, and ensures compatibility with any pretrained monocular RGB normal estimator in a plug-and-play fashion, independent of its architecture.

4.2 Polarization-guidance parameters

To correct the erroneous normal $\hat{n}_{\text{base}}$ while keeping the model weights frozen, we introduce learnable parameters applied before and after f. Formally, the refined normal at step t is expressed as $\hat{n}_t = g_{\text{a}}(f(g_{\text{b}}(x)))$, where g_{b} and g_{a} denote arbitrary functions applied to the input and output of f, respectively. We adopt the simplest instantiation, $g_{\text{b}}(x) = x + O_x$ and $g_{\text{a}}(\hat{n}_{\text{base}}) = \hat{n}_{\text{base}} + O_n$, yielding $\hat{n}_t = f(x + O_x) + O_n$, where $O_x \in \mathbb{R}^{H\times W\times 3}$ and $O_n \in \mathbb{R}^{H\times W\times 3}$ are per-pixel offsets applied to the input image x and the predicted normal $\hat{n}_{\text{base}}$, respectively.

Global Guidance. The image offset O_x is motivated by the observation that even a small perturbation in input space can produce a globally significant correction in the output normals. This behavior is explained by the Jacobian of different models f, visualized in Fig. 4(b), which shows that a change in a single input pixel has a broad influence over the output normal map (details in Sec. A in the Supplement). Consequently, the image offset O_x serves as an efficient mechanism for globally steering the predicted geometry toward a polarization-consistent solution.

Local Guidance. The normal offset O_n, applied directly to the output, addresses the tendency of monocular normal estimators to produce smooth predictions. The normal offset O_n recovers high-frequency surface details that f fails to capture. Due to its direct coupling with the normal output, the normal offset O_n exhibits high sensitivity to the polarization loss. We therefore defer its optimization until L_s has converged sufficiently to yield a reliable estimate of Stokes $\hat{\mathbf{S}} = \mathcal{F}(f(x+O_x)+O_n, L_s)$ (third column in Fig. 4(a)). We assume orthographic projection and ignore viewing directions, an approximation that introduces negligible error unless the object is captured at close range.



(a) Effect of joint optimization on Stokes reconstruction (b) Input-to-normal sensitivity (Jacobian)

Fig. 4: (a) Stokes reconstruction of hedgehog scene (from NeRSP) when: $L_s = 0$ (diffuse only) in column 1; only L_s is learned from backbone predicted normals $\hat{n}_{\text{base}}$ in column 2; L_s, O_x, and O_n are jointly learned in column 3. (b) Jacobian magnitude maps for a selected input pixel, normalized by the 99th percentile for different backbones, showing how perturbations of a single input pixel (green dot) influence the output normal map at a global level, across spatial locations.

4.3 Guidance objective and updates

During test-time guidance, we convert the refined normal $\hat{n}_t$ into Stokes $\hat{\mathbf{S}}$ for polarization consistency check. We optimize the learnable parameters $\Theta = \{L_s, O_x, O_n\}$ iteratively over T steps to minimize the discrepancy between the observed and predicted Stokes vectors. At each step t, the backbone f takes the offset input $x+O_x$ and produces the base normal $\hat{n}_{\text{base}} = f(x+O_x)$. The normal offset O_n is then applied to yield the refined normal $\hat{n}_t = \hat{n}_{\text{base}} + O_n$. Given $\hat{n}_t$ and the specular radiance L_s, the predicted Stokes vector $\hat{\mathbf{S}}(\hat{n}_t, L_s)$ is computed following Sec. 3.2. The polarization consistency loss is defined as:

$$\mathcal{L} = \sum_p M(p) \sum_{i=0}^{2} \left| S_i(p) - \widehat{S}_i(p) \right|, \tag{3}$$

where p denotes a pixel and $M(p)$ is a validity mask that restricts the loss to physically meaningful measurements. Specifically, a pixel is included if it has sufficient signal ($S_0 > 0.01$), is not saturated ($S_0 < 1$), and satisfies the physical Stokes constraint ($S_1^2 + S_2^2 \leq S_0^2$).

Global-then-local guidance. The three parameters are optimized in a staged schedule, each with a distinct learning rate. In the early phase ($t < 50$), we only update L_s and O_x, allowing the specular radiance to converge and the predicted normals to reach global geometric plausibility. At $t = 50$, O_n is introduced to refine high-frequency surface details that the backbone fails to capture, and all three parameters are subsequently updated until $t = T$.

The learnable parameters $\Theta \in \{L_s, O_x, O_n\}$ are updated with an optimizer using gradient descent $\Theta^{(k+1)} = \Theta^{(k)} - \lambda_\Theta \nabla_\Theta \mathcal{L}$. After T iterations, the output surface normal is $\hat{n} = \frac{f(x+O_x)+O_n}{\|f(x+O_x)+O_n\|_2}$.

Table 1: Aggregated normal estimation performance on real and synthetic benchmarks. Our polarization guidance (+ Poppy) consistently reduces MAE and RMSE across backbone RGB estimators while improving thresholded accuracy, confirming gains in both global surface orientation and fine-scale local geometric detail. +Poppy denotes joint optimization of the image offset and normal offset. Best result per backbone within each data split is highlighted in bold.

	Real						**Synthetic**					
Method	Mean	Median	RMSE	Acc11.25	Acc22.5	Acc30	Mean	Median	RMSE	Acc11.25	Acc22.5	Acc30
SfPUEL [39]	20.68	17.08	26.15	0.37	0.70	0.80	17.03	12.91	23.33	0.46	0.79	0.87
DSINE [6]	23.06	19.07	28.83	0.27	0.62	0.76	24.13	19.94	30.00	0.26	0.60	0.75
Lotus [27]	16.69	13.96	21.70	0.42	0.78	0.88	19.99	17.04	25.06	0.30	0.69	0.84
StableNormal [55]	17.80	14.77	23.29	0.41	0.75	0.86	18.45	14.65	24.55	0.39	0.75	0.86
Marigold [34]	18.18	15.25	23.30	0.36	0.74	0.86	20.99	17.40	26.43	0.30	0.68	0.82
Marigold + Poppy	**15.28**	**12.57**	**20.26**	**0.47**	**0.83**	**0.92**	**15.60**	**11.89**	**22.08**	**0.56**	**0.82**	**0.89**
Lotus-v2 [26]	14.68	12.05	19.51	0.49	0.85	0.93	16.52	13.44	22.08	0.42	0.81	0.90
Lotus-v2 + Poppy	**12.65**	**10.05**	**17.62**	**0.61**	**0.89**	**0.94**	**12.26**	**8.81**	**18.70**	**0.66**	**0.89**	**0.93**
MoGe-2 [52]	13.10	10.51	18.11	0.57	0.87	0.94	14.13	11.33	20.01	0.51	0.87	0.94
MoGe-2 + Poppy	**12.26**	**9.88**	**16.99**	**0.61**	**0.90**	**0.95**	**10.89**	**8.05**	**16.41**	**0.70**	**0.92**	**0.96**

The learned specular radiance L_s and the refined output normal $\hat{n}$ together synthesize the predicted Stokes vector $\hat{\mathbf{S}}$ that closely matches the observed Stokes vector $\mathbf{S}$ as shown in the third column in Fig. 4(a). Consequently, minimizing this discrepancy drives the predicted normal $\hat{n}$ toward the true surface normal.

5 Results

5.1 Implementation details

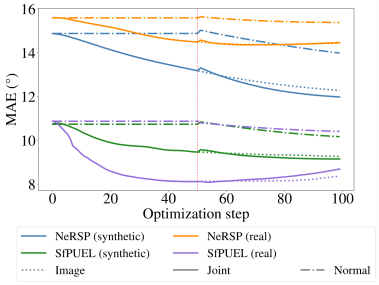
Datasets. Synthetic benchmarks include SfPUEL [39] (low SNR, highly specular), NeRSP [25] (high reflectance, low illumination), NeISF [36] (moderate reflectance), and DeepPol [16] (mixed diffuse and specular). Real-world benchmarks include SfPUEL [39], NeRSP [25], and PISR [8] (highly reflective, textureless), with ground-truth normals extracted from provided 3D meshes.

Baseline models. We compare against the polarization-based model SfPUEL [39] and RGB-based methods DSINE [6], StableNormal [55], Marigold [34], Lotus-v2 [26], and MoGe-2 [52], all under their default configurations. Poppy is applied to Marigold, Lotus-v2, and MoGe-2, representing diffusion-based, flow-based, and feed-forward backbones. We report mean angular error (MAE), median angular error (Median), angular RMSE, and accuracy under angular thresholds.

Optimization. We optimize with Adam for 100 steps. Learning rates: $\lambda_{L_s} = 0.01$; backbone-dependent rates for O_x: 10^{-4} (Marigold), 5×10^{-4} (Lotus-v2), 10^{-5} (MoGe-2). The normal offset O_n uses $\lambda_n = 0.001$ across all backbones and activates at step 50, once L_s has converged.

Backbone	Variant	Synthetic				Real		
		SfPUEL	NeRSP	NeISF	DeepPol	SfPUEL	NeRSP	PISR
Marigold	None	15.19	21.47	20.03	27.29	17.45	19.03	18.06
	Image	12.71	16.92	12.67	21.35	**12.01**	**16.98**	16.85
	Joint	**12.45**	**16.55**	**12.33**	**21.07**	12.12	17.01	**16.71**
Lotus-v2	None	13.14	17.85	15.30	19.80	12.38	17.76	13.90
	Image	10.25	14.17	10.19	15.26	**9.08**	**15.37**	**12.78**
	Joint	**10.22**	**13.82**	**9.92**	**15.08**	9.64	15.48	12.82
MoGe-2	None	10.93	16.08	13.55	15.95	10.84	15.63	**12.82**
	Image	9.31	12.27	9.98	13.04	**8.42**	**14.49**	13.80
	Joint	**9.19**	**11.98**	**9.58**	**12.80**	8.74	14.51	13.54

(a) MAE on synthetic and real-world datasets



(b) Effect of normal offset activation

Fig. 5: Ablation of guidance variants on real and synthetic datasets. (a) Mean angular error (MAE) across three backbones (Marigold, Lotus-v2, MoGe-2) with no guidance (None), image offset guidance (Image), and joint image and normal offsets guidance (Joint). Guidance improves normals, with joint guidance performing slightly better on synthetic datasets. (b) MAE over optimization steps on SfPUEL and NeRSP with MoGe-2 backbone. The red-dashed line at t=50 marks the activation of the normal offset. On synthetic data, joint guidance accelerates error reduction. On real data, sensor noise causes the normal offset to slightly increase MAE, though high-frequency detail is still recovered.

Backbone inference. For Marigold, we use 4 denoising steps with 25 guidance steps each, without ensemble inference. For Lotus-v2, guidance precedes the detail-sharpening stage, followed by 10 sharpening iterations at default settings. For MoGe-2, gradients are backpropagated directly through the feed-forward pipeline. See Sec. C in the Supplement for per-backbone details.

Runtime and memory. At 768 × 768 resolution, per-step cost splits roughly 45/55 between inference and optimization. Per-step costs: 567 ms (MoGe-2), 950 ms (Marigold), 1729 ms (Lotus-v2), yielding total runtimes of 57 s, 99 s, and 173 s at 100 steps. Peak memory: 12.97 GB (MoGe-2), 35.13 GB (Marigold), 66.64 GB (Lotus-v2), dominated by gradient retention during optimization. All experiments run on NVIDIA RTX PRO 6000 Blackwell GPUs (95.6 GB VRAM).

5.2 Results and analysis

Quantitative comparisons with monocular normal baselines. Tab. 1 reports aggregated performance across all backbones and benchmarks. Adding polarization guidance (+Poppy) consistently improves MAE on both real and synthetic datasets, with MAE reductions of 6–26% and RMSE reductions of 7–18%, reflecting improved global surface orientation consistency. More notably, Acc11.25 (the fraction of pixels with angular error below 11.25°) improves by 7–31% on real data and 37–87% on synthetic data, demonstrating that Poppy not only corrects large-scale geometric errors but also recovers high-frequency surface details that RGB-based backbones systematically fail to capture.

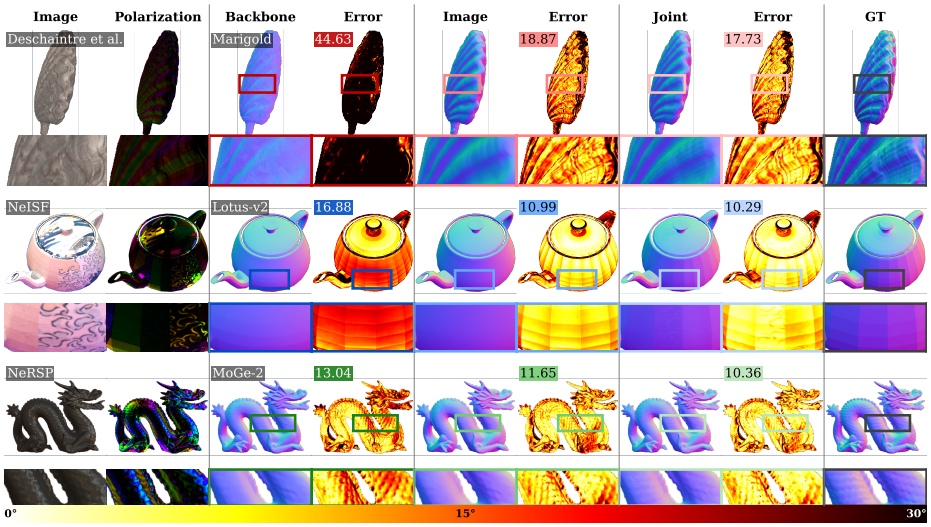


Fig. 6: Comparison of image (global) and joint (global-then-local) guidance on synthetic data. We compare output normals under three guidance conditions across three backbones: no guidance (Backbone), image offsets (Image), and joint image and normal offsets (Joint). Joint guidance recovers fine geometric structures – shell ridges, mesh facets, and dragon scales – that image-offset guidance alone cannot resolve. Pixel-wise angular error maps (brighter is lower error) and mean angular error are also shown for each prediction. Best viewed at high magnification.

Qualitative results on real-world scenes. To illustrate how these aggregate gains manifest in individual scenes, Fig. 7 presents qualitative comparisons on real-world scenes that are particularly challenging for RGB-based estimators. These scenes include textureless low-albedo surfaces, highly specular objects, and low-SNR conditions arising from low illumination, low albedo, or high measurement noise. Across all three backbone architectures and all challenging conditions, Poppy yields MAE reductions of 19–41% over the backbone predictions, with the largest improvements on scenes where RGB-based appearance cues alone are most ambiguous.

Effect of global and local guidance. Fig. 5 and Fig. 6 analyze the individual contributions of the image offset O_x and the normal offset O_n. In Fig. 5(a), the image offset O_x alone accounts for the majority of the MAE reduction across all backbones and datasets, correcting large-scale geometric errors toward a globally polarization-consistent solution. Adding the normal offset O_n (Joint) yields a further reduction on synthetic data, though the additional gain is modest compared to the image offset alone. Across all backbones and datasets, Poppy outperforms the backbone baseline, with the sole exception of MoGe-2 on PISR, which we attribute to pronounced perspective distortion that our orthographic

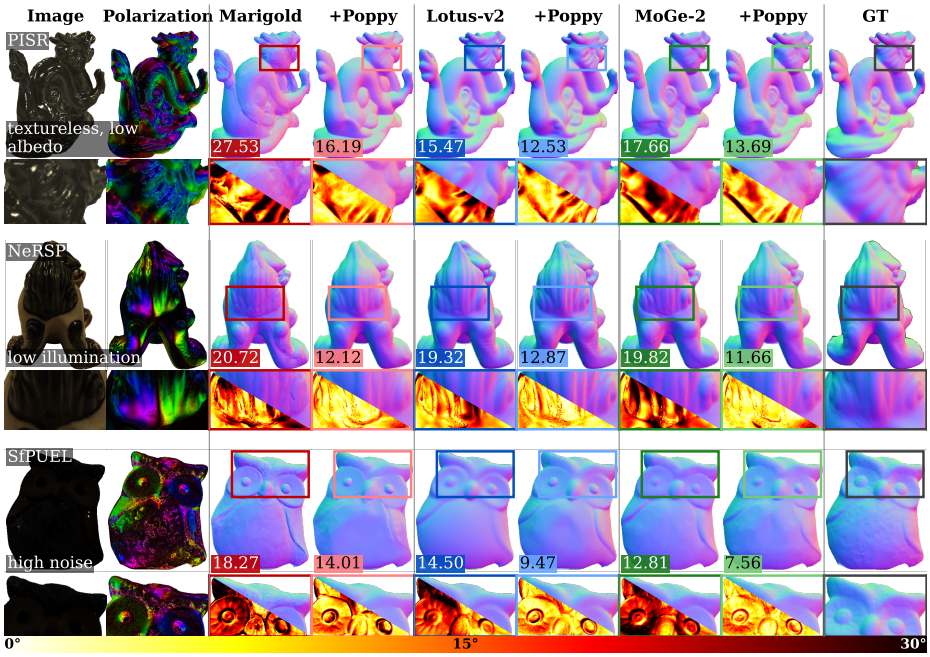
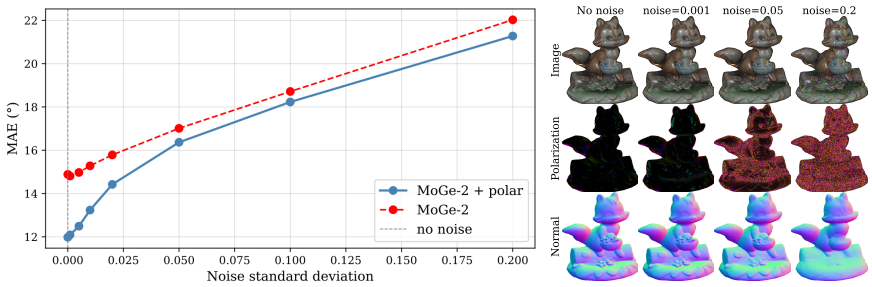


Fig. 7: Qualitative comparison on real-world scenes. Columns show backbone predictions and results after Poppy guidance. Pixel-wise angular error maps accompany each prediction (brighter is lower error), with the mean angular error annotated. Scenes span three challenging categories: textureless and highly reflective objects, and low-SNR conditions (low illumination, low albedo, high noise). Poppy shows consistent improvement in normal quality across all categories and backbones.

assumption does not model. See Sec. D in the Supplement for perspective-aware guidance by incorporating field-of-view.

The per-step MAE curve in Fig. 5(b) is consistent with this global-then-local behavior: MAE decreases primarily during the early optimization phase under the image offset; at t=50, when the normal offset is activated on synthetic data, a further sharp drop occurs as joint offset learning refines the remaining errors. Once jointly optimized (Fig. 6), the normal offset recovers fine-scale geometric structures that the backbone over-smooths, such as shell ridges, mesh facets, and dragon scales (zoomed insets).

Robustness to sensor noise. We examine how Poppy behaves when the polarization input itself is corrupted. Fig. 8 reports MAE after adding zero-mean Gaussian noise of varying standard deviation σ to the NeRSP synthetic polarization measurements. At low σ, Poppy reduces MAE relative to the no-guidance prediction on MoGe-2. As σ grows, the guided result converges toward the unguided backbone prediction – noisy cues are effectively ignored rather than amplified.



(a) Normal estimation with varying noise levels (b) Qualitative results with increasing noise level

Fig. 8: Sensitivity to input noise. We add zero-mean Gaussian noise of increasing standard deviation σ to the NeRSP synthetic measurements. The plot reports mean angular error (MAE) for MoGe-2 with and without polarization guidance. Guidance improves accuracy at low noise levels, but the advantage diminishes with increasing σ.

Robustness to material properties. We isolate the effect of material reflectance using the NeISF synthetic dataset (Fig. 9), testing three conditions: purely diffuse, purely specular, and mixed reflectance. MoGe-2 without guidance achieves comparable MAE across all three conditions, but polarization guidance (+ Poppy) shows the most refinement on the specular case, where specular reflection produces stronger linear polarization. Diffuse surfaces yield the second-largest improvement, as even weaker polarization from diffuse radiance carries useful orientation information. In the mixed case, accurate diffuse-specular separation becomes a prerequisite; errors in this decomposition propagate into the polarization guidance, resulting in smaller but still meaningful gains.

5.3 Applications

Appearance decomposition. Beyond refining normals, Poppy yields a physics-based decomposition of captured appearance (S_0) into diffuse and specular components (L_d and L_s), as shown in Fig. 2 (top row). We provide image editing applications in Sec. H in the Supplement. From the decomposed appearance and refined normals, our forward model synthesizes the polarization properties of diffuse and specular components (Fig. 2 bottom row).

3D mesh reconstruction. In Fig. 10, we test our improved normals on downstream mesh reconstruction tasks. We use VCR-GauS [9] as our baseline to construct surface mesh, guided by multi-view RGB images and monocular surface normal estimates on those views. We observe an improvement in reconstructed mesh accuracy with our surface normals and an ≈6% improvement in Chamfer distance as compared to RGB-only backbones. Note that we attempted using Gaussian Surfels [14]. However, it failed due to poor initialization points retrieved by COLMAP [44, 45] on glossy, textureless surfaces.

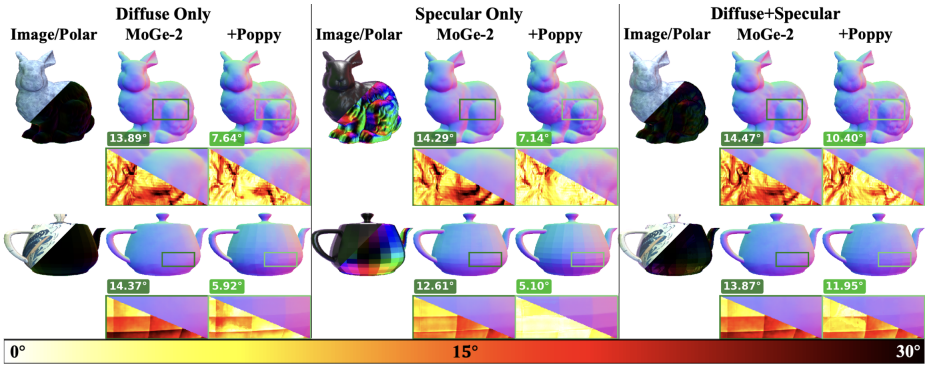


Fig. 9: Robustness to material reflectance properties. Using the NeISF dataset, we render scenes with purely specular, purely diffuse, and mixed diffuse and specular reflectance to evaluate polarization guidance under varying material properties. Specular reflectance produces the strongest polarization signal and yields the largest improvement, followed by diffuse reflectance. Mixed reflectance case yields smaller gains due to the challenge of accurately decomposing diffuse and specular components. It shows pixel-wise angular error maps (brighter is lower error) with the mean angular error annotated.

6 Conclusion

We introduced Poppy, a training-free framework that guides frozen RGB normal estimators with physics-based polarization constraints at test time, refining normals on reflective, textureless, and dark surfaces without polarimetric training data or backbone retraining. A learnable image offset propagates global corrections through the backbone; a normal offset recovers high-frequency local surface detail; and a learned specular radiance map resolves diffuse–specular ambiguity in single-view polarization. Because guidance parameters lie entirely outside the backbone, Poppy applies to diffusion-based [34], flow-based [26], and feed-forward [52] backbones in a plug-and-play manner. Across seven benchmarks, the framework reduces mean angular error by 23–26% on synthetic data and 6–16% on real captures.

Although we focus on showing that polarization can guide any frozen monocular estimator at test time without retraining, our framework opens several directions for future work. The iterative optimization requires multiple forward passes through the backbone; however, we expect that better optimization strategies can reduce the computational overhead. The learnable image and normal offsets are lightweight yet effective for normal refinement; replacing them with neural network adapters could further increase the representation capacity at the expense of requiring minimal polarization data for fine-tuning. Extending robustness to higher sensor noise levels, handling perspective distortions for close-range captures, and modeling more complex material types such as metallic, transparent, and translucent surfaces are all within reach of the modular framework. We

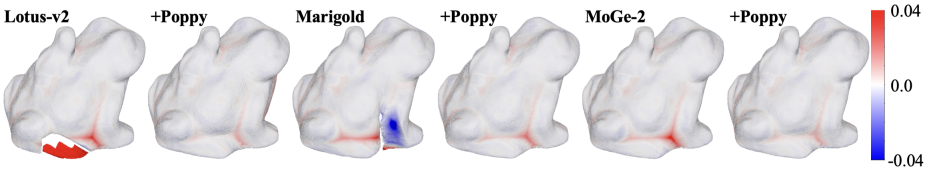


Fig. 10: Mesh reconstruction with refined surface normals. We reconstruct meshes using VCR-GauS [9] from multi-view images and corresponding normals estimated with and without polarization guidance. Polarization-refined normals yield better mesh quality across the three backbones. Signed distance to the ground-truth mesh is shown for each reconstruction.

see this work as a step toward steering frozen foundation models with physics-based cues, available solely at test time.

References

1. Atkinson, G.A.: Polarisation photometric stereo. Computer Vision and Image Understanding **160**, 158–167 (2017)
2. Atkinson, G.A., Hancock, E.R.: Recovery of surface orientation from diffuse polarization. IEEE transactions on image processing **15**(6), 1653–1664 (2006)
3. Atkinson, G.A., Hancock, E.R.: Shape estimation using polarization and shading from two views. IEEE transactions on pattern analysis and machine intelligence **29**(11), 2001–2017 (2007)
4. Atkinson, G.A., Hancock, E.R.: Surface reconstruction using polarization and photometric stereo. In: International conference on computer analysis of images and patterns. pp. 466–473. Springer (2007)
5. Ba, Y., Gilbert, A., Wang, F., Yang, J., Chen, R., Wang, Y., Yan, L., Shi, B., Kadambi, A.: Deep shape from polarization. In: European Conference on Computer Vision. pp. 554–571. Springer (2020)
6. Bae, G., Davison, A.J.: Rethinking inductive biases for surface normal estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9535–9545 (2024)
7. Baek, S.H., Jeon, D.S., Tong, X., Kim, M.H.: Simultaneous acquisition of polarimetric svbrdf and normals. ACM Trans. Graph. **37**(6), 268 (2018)
8. Chen, G., He, Y., He, L., Zhang, H.: Pisr: Polarimetric neural implicit surface reconstruction for textureless and specular objects. In: European Conference on Computer Vision. pp. 205–222. Springer (2024)
9. Chen, H., Wei, F., Li, C., Huang, T., Wang, Y., Lee, G.H.: Vcr-gaus: View consistent depth-normal regularizer for gaussian surface reconstruction. Advances in Neural Information Processing Systems **37**, 139725–139750 (2024)
10. Chen, Z., Pan, Y., Ye, Y., Lu, M., Xia, Y.: Each test image deserves a specific prompt: Continual test-time adaptation for 2d medical image segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11184–11193 (2024)
11. Chung, H., Kim, J., Mccann, M.T., Klasky, M.L., Ye, J.C.: Diffusion posterior sampling for general noisy inverse problems. arXiv preprint arXiv:2209.14687 (2022)

12. Collett, E.: Field guide to polarization, vol. 15. SPIE press Bellingham, Washington (2005)
13. Cui, Z., Gu, J., Shi, B., Tan, P., Kautz, J.: Polarimetric multi-view stereo. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1558–1567 (2017)
14. Dai, P., Xu, J., Xie, W., Liu, X., Wang, H., Xu, W.: High-quality surface reconstruction using gaussian surfels. In: ACM SIGGRAPH 2024 conference papers. pp. 1–11 (2024)
15. Dave, A., Zhao, Y., Veeraraghavan, A.: Pandora: Polarization-aided neural decomposition of radiance. In: European conference on computer vision. pp. 538–556. Springer (2022)
16. Deschaintre, V., Lin, Y., Ghosh, A.: Deep polarization imaging for 3d shape and svbrdf acquisition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15567–15576 (2021)
17. Ding, Y., Ji, Y., Zhou, M., Kang, S.B., Ye, J.: Polarimetric helmholtz stereopsis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5037–5046 (2021)
18. Do, T., Vuong, K., Roumeliotis, S.I., Park, H.S.: Surface normal estimation of tilted images via spatial rectifier. In: European Conference on Computer Vision. pp. 265–280. Springer (2020)
19. Efron, B.: Tweedie's formula and selection bias. Journal of the American Statistical Association **106**(496), 1602–1614 (2011)
20. Eftekhar, A., Sax, A., Malik, J., Zamir, A.: Omnidata: A scalable pipeline for making multi-task mid-level vision datasets from 3d scans. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10786–10796 (2021)
21. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: Proceedings of the IEEE international conference on computer vision. pp. 2650–2658 (2015)
22. Fu, X., Yin, W., Hu, M., Wang, K., Ma, Y., Tan, P., Shen, S., Lin, D., Long, X.: Geowizard: Unleashing the diffusion priors for 3d geometry estimation from a single image. In: European Conference on Computer Vision. pp. 241–258. Springer (2024)
23. Garcia, G.M., Zeid, K.A., Schmidt, C., De Geus, D., Hermans, A., Leibe, B.: Fine-tuning image-conditional diffusion models is easier than you think. In: Proceedings of the Winter Conference on Applications of Computer Vision. pp. 753–762 (2025)
24. Graikos, A., Jojic, N., Samaras, D.: Fast constrained sampling in pre-trained diffusion models. arXiv preprint arXiv:2410.18804 (2024)
25. Han, Y., Guo, H., Fukai, K., Santo, H., Shi, B., Okura, F., Ma, Z., Jia, Y.: Nersp: Neural 3d reconstruction for reflective objects with sparse polarized images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11821–11830 (2024)
26. He, J., Li, H., Sheng, M., Chen, Y.C.: Lotus-2: Advancing geometric dense prediction with powerful image generative model. arXiv preprint arXiv:2512.01030 (2025)
27. He, J., Li, H., Yin, W., Liang, Y., Li, L., Zhou, K., Zhang, H., Liu, B., Chen, Y.C.: Lotus: Diffusion-based visual foundation model for high-quality dense prediction. arXiv preprint arXiv:2409.18124 (2024)
28. Huynh, C.P., Robles-Kelly, A., Hancock, E.R.: Shape and refractive index from single-view spectro-polarimetric images. International journal of computer vision **101**(1), 64–94 (2013)

29. Hyoseok, L., Kim, K.S., Byung-Ki, K., Oh, T.H.: Zero-shot depth completion via test-time alignment with affine-invariant depth prior. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 3877–3885 (2025)
30. Ichikawa, T., Purri, M., Kawahara, R., Nobuhara, S., Dana, K., Nishino, K.: Shape from sky: Polarimetric normal recovery under the sky. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14832–14841 (2021)
31. Ikemura, K., Huang, Y., Heide, F., Zhang, Z., Chen, Q., Lei, C.: Robust depth enhancement via polarization prompt fusion tuning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 20710–20720 (2024)
32. Jeong, C., Bae, I., Park, J.H., Jeon, H.G.: Test-time prompt tuning for zero-shot depth completion. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9443–9454 (2025)
33. Kadambi, A., Taamazyan, V., Shi, B., Raskar, R.: Polarized 3d: High-quality depth sensing with polarization cues. In: Proceedings of the IEEE international conference on computer vision. pp. 3370–3378 (2015)
34. Ke, B., Obukhov, A., Huang, S., Metzger, N., Daudt, R.C., Schindler, K.: Repurposing diffusion-based image generators for monocular depth estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9492–9502 (2024)
35. Lei, C., Qi, C., Xie, J., Fan, N., Koltun, V., Chen, Q.: Shape from polarization for complex scenes in the wild. In: Proceedings of the ieee/cvf conference on computer vision and pattern recognition. pp. 12632–12641 (2022)
36. Li, C., Ono, T., Uemori, T., Mihara, H., Gatto, A., Nagahara, H., Moriuchi, Y.: Neisf: Neural incident stokes field for geometry and material estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21434–21445 (2024)
37. Li, Z., Zhong, Z., Nobuhara, S., Nishino, K., Zheng, Y.: Fooling polarization-based vision using locally controllable polarizing projection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 24706–24715 (2024)
38. Long, X., Guo, Y.C., Lin, C., Liu, Y., Dou, Z., Liu, L., Ma, Y., Zhang, S.H., Habermann, M., Theobalt, C., et al.: Wonder3d: Single image to 3d using cross-domain diffusion. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9970–9980 (2024)
39. Lyu, Y., Guo, H., Zhang, K., Li, S., Shi, B.: Sfpuel: Shape from polarization under unknown environment light. Advances in Neural Information Processing Systems **37**, 97184–97202 (2024)
40. Miyazaki, Tan, Hara, Ikeuchi: Polarization-based inverse rendering from a single view. In: Proceedings Ninth IEEE International Conference on Computer Vision. pp. 982–987. IEEE (2003)
41. Morel, O., Meriaudeau, F., Stolz, C., Gorria, P.: Polarization imaging applied to 3d reconstruction of specular metallic surfaces. In: Machine Vision Applications in Industrial Inspection XIII. vol. 5679, pp. 178–186. SPIE (2005)
42. Ngo Thanh, T., Nagahara, H., Taniguchi, R.i.: Shape and light directions from shading and polarization. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2310–2318 (2015)
43. Qi, X., Liu, Z., Liao, R., Torr, P.H., Urtasun, R., Jia, J.: Geonet++: Iterative geometric neural network with edge-aware refinement for joint depth and surface

normal estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(2), 969–984 (2020)
44. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4104–4113 (2016)
45. Schönberger, J.L., Zheng, E., Frahm, J.M., Pollefeys, M.: Pixelwise view selection for unstructured multi-view stereo. In: European conference on computer vision. pp. 501–518. Springer (2016)
46. Smith, W.A., Ramamoorthi, R., Tozza, S.: Height-from-polarisation with unknown lighting or albedo. IEEE transactions on pattern analysis and machine intelligence **41**(12), 2875–2888 (2018)
47. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models (2020)
48. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199 (2013)
49. Talegaonkar, C., Suresh, N.G., Novack, Z., Belhe, Y., Nagasamudra, P., Antipa, N.: Repurposing marigold for zero-shot metric depth estimation via defocus blur cues. arXiv preprint arXiv:2505.17358 (2025)
50. Tozza, S., Smith, W.A., Zhu, D., Ramamoorthi, R., Hancock, E.R.: Linear differential constraints for photo-polarimetric height estimation. In: Proceedings of the IEEE international conference on computer vision. pp. 2279–2287 (2017)
51. Viola, M., Qu, K., Metzger, N., Ke, B., Becker, A., Schindler, K., Obukhov, A.: Marigold-dc: Zero-shot monocular depth completion with guided diffusion (2025)
52. Wang, R., Xu, S., Dong, Y., Deng, Y., Xiang, J., Lv, Z., Sun, G., Tong, X., Yang, J.: Moge-2: Accurate monocular geometry with metric scale and sharp details (2025)
53. Xu, G., Ge, Y., Liu, M., Fan, C., Xie, K., Zhao, Z., Chen, H., Shen, C.: What matters when repurposing diffusion models for general dense perception tasks? arXiv preprint arXiv:2403.06090 (2024)
54. Yang, L., Tan, F., Li, A., Cui, Z., Furukawa, Y., Tan, P.: Polarimetric dense monocular slam. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3857–3866 (2018)
55. Ye, C., Qiu, L., Gu, X., Zuo, Q., Wu, Y., Dong, Z., Bo, L., Xiu, Y., Han, X.: Stablenormal: Reducing diffusion variance for stable and sharp normal. ACM Transactions on Graphics (ToG) **43**(6), 1–18 (2024)
56. Yu, J., Wang, Y., Zhao, C., Ghanem, B., Zhang, J.: Freedom: Training-free energy-guided conditional diffusion model. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 23174–23184 (2023)

A Visualizing input-to-normal sensitivity using Jacobians

Here we describe the visualization used in Fig. 4(b) and Sec. 4.2 to characterize input-to-normal sensitivity of monocular normal estimators. The model f maps an input RGB image $I \in \mathbb{R}^{3\times H\times W}$ to a surface normal map $n \in \mathbb{R}^{H\times W\times 3}$, where the output channels correspond to the normal components $n[y, x] = (n_x, n_y, n_z)$.

We analyze how perturbing a single input pixel affects the predicted normals across the entire output image. Let the chosen input pixel location be (p_i, p_j). Our goal is to measure how sensitive each output pixel (i', j') is to small perturbations of this input pixel.

Step 1: Computing Jacobian–Vector Products. The full Jacobian of the network is $J_f(I) = \frac{\partial n}{\partial I}$, which maps perturbations in the input image to perturbations in the output normals. For each input color channel $c_k \in \{R, G, B\}$, we construct a tangent vector $v^{(c_k)} \in \mathbb{R}^{3\times H\times W}$, defined as

$$v^{(c_k)}_{i,j,c} = \begin{cases} 1 & \text{if } i = p_i,\ j = p_j,\ c = c_k \\ 0 & \text{otherwise.} \end{cases}$$

This vector represents an infinitesimal perturbation of a single input variable $I[c_k, p_i, p_j]$. Using forward-mode automatic differentiation, we compute the Jacobian–vector product $\mathrm{JVP}^{(c_k)} = J_f(I)\, v^{(c_k)}$. Because $v^{(c_k)}$ is a one-hot vector in the input space, the JVP extracts exactly one column of the full Jacobian:

$$\mathrm{JVP}^{(c_k)}[c_o, i', j'] = \frac{\partial n[c_o, i', j']}{\partial I[c_k, p_i, p_j]},$$

where $c_o \in \{n_x, n_y, n_z\}$ denotes the output normal component. Repeating this procedure for the three input channels provides all partial derivatives relating the selected input pixel to the output normals.

Step 2: Local Jacobian Block. For a fixed output pixel (i', j'), we assemble a 3×3 local Jacobian block

$$J_{i',j'} = \begin{pmatrix} \frac{\partial n_x[i',j']}{\partial I[R,p_i,p_j]} & \frac{\partial n_x[i',j']}{\partial I[G,p_i,p_j]} & \frac{\partial n_x[i',j']}{\partial I[B,p_i,p_j]} \\ \frac{\partial n_y[i',j']}{\partial I[R,p_i,p_j]} & \frac{\partial n_y[i',j']}{\partial I[G,p_i,p_j]} & \frac{\partial n_y[i',j']}{\partial I[B,p_i,p_j]} \\ \frac{\partial n_z[i',j']}{\partial I[R,p_i,p_j]} & \frac{\partial n_z[i',j']}{\partial I[G,p_i,p_j]} & \frac{\partial n_z[i',j']}{\partial I[B,p_i,p_j]} \end{pmatrix}.$$

Each column of this matrix corresponds to the JVP result for a particular input channel.

Step 3: Frobenius Norm of the Local Jacobian. To summarize the total sensitivity of output pixel (i', j') to perturbations of the selected input pixel, we compute the Frobenius norm of this local Jacobian block:

$$\|J_{i',j'}\|_F = \sqrt{\sum_{c_k \in \{R,G,B\}} \sum_{c_o \in \{n_x, n_y, n_z\}} \left(\frac{\partial n[c_o, i', j']}{\partial I[c_k, p_i, p_j]} \right)^2}.$$

Interpretation. The value $\|J_{y,x}\|_F$ measures the overall sensitivity of the predicted normal at pixel (i', j') to perturbations of the input pixel (p_i, p_j), aggregated across all input and output channels. Pixels that are weakly influenced by the selected input location will have $\|J_{y,x}\|_F \approx 0$, while pixels strongly coupled through the network will exhibit larger values. This Jacobian norm therefore provides a spatial map describing how perturbations at a single input pixel propagate through the model to affect the predicted surface normals globally as described in Sec. 4.2. The Jacobian magnitude map for a single-pixel perturbation is shown in Fig. 4(b).

B Effect of image guidance on the network input

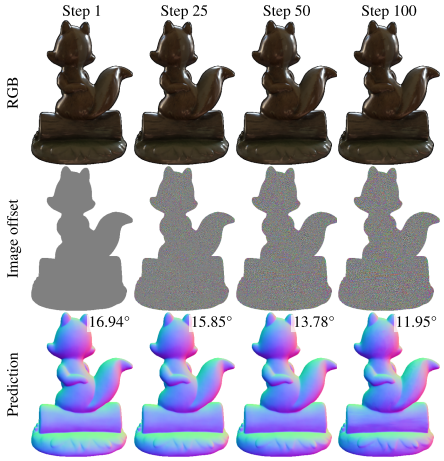


Fig. 11: Test-time image optimization behavior. The input image is iteratively refined using a learned offset. The image offset is multiplied by 50 for visualization. Although the perturbations appear as nearly imperceptible noise in the RGB image, it produces large changes in the predicted surface normals. The optimization progressively improves the normal prediction, reducing the mean angular error from 16.94° to 11.95°.

Fig. 11 illustrates the effect of the proposed test-time image optimization. Although the image offset changes the input image only slightly during optimization, the predicted surface normals show substantial improvement. The image

offset appears as small high-frequency noise in the image space and is nearly imperceptible to human eyes. Nevertheless, these small perturbations significantly change the network output, reducing the mean angular error from 16.94° to 11.95°.

This phenomenon resembles adversarial perturbations [48] in deep networks. In adversarial examples, carefully designed small perturbations can dramatically alter the network prediction while remaining visually imperceptible. In our setting, however, the perturbation is not used to degrade the model but instead to guide the network toward a physically consistent solution. The image offset effectively acts as small perturbations that exploit the model's sensitivity to correct geometric errors while preserving the visual appearance of the input image.

C Backbone inference details

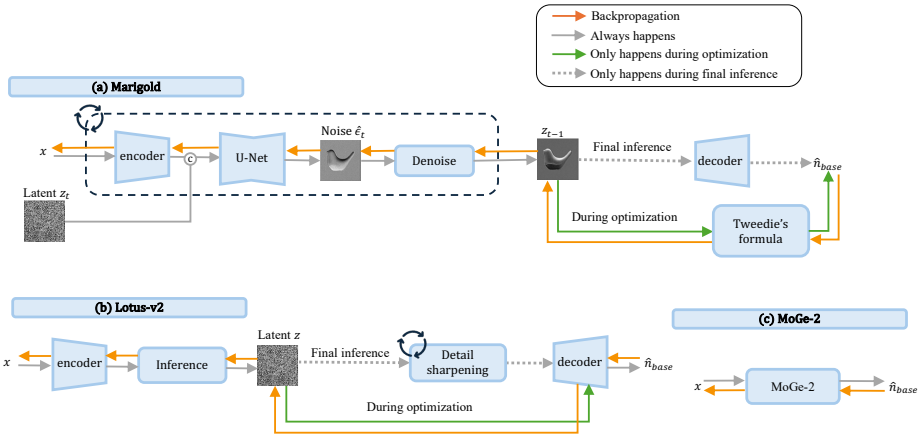


Fig. 12: Optimization details for different backbone architectures in Poppy. (a) Marigold performs optimization at each denoising step using the normal preview obtained via Tweedie's formula. (b) Lotus-v2 performs optimization during the latent inference stage before the detail sharpening step. (c) MoGe-2 is a feed-forward network, allowing gradients to directly propagate to the input image during optimization.

The proposed Poppy optimization framework is compatible with different types of monocular normal estimation models. In Sec. 5.1, we evaluate three representative backbones with distinct inference pipelines: a diffusion-based model (Marigold [34]), a multi-stage flow-based model (Lotus-v2 [26]), and a feed-forward model (MoGe-2 [52]), as illustrated in Fig. 12.

Marigold predicts surface normals through an iterative denoising process. As shown in Fig. 12(a), multiple optimization steps are performed at each denoising

step. At every DDIM [47] step, a normal preview is computed using Tweedie's formula [19] from the predicted noise, and the polarization loss is evaluated on this preview. Rather than backpropagating through the entire diffusion trajectory, optimization is applied locally at each denoising step. This significantly reduces memory consumption and allows Poppy to accommodate arbitrary numbers of denoising steps selected at inference time.

Lotus-v2 follows a multi-stage inference pipeline consisting of latent inference, latent refinement (detail sharpening), and decoding, as illustrated in Fig. 12(b). Poppy optimization is applied during the latent inference stage. The model first predicts a latent representation from the input image, which is then decoded to obtain the surface normal prediction used to compute the polarization loss.

Although the subsequent sharpening stage refines the latent representation, it operates only on the latent variables and does not directly depend on the input image. Therefore, the optimization focuses on the latent inference stage, where the image influences the prediction. As in the Marigold case, we avoid backpropagating through the entire inference pipeline because it would be memory-inefficient, especially when multiple sharpening steps are used. Instead, the optimization updates the image based on the latent prediction produced during the initial inference step. After optimization, the resulting latent representation continues through the remaining refinement and decoding stages during inference. This design allows Poppy to remain memory-efficient while accommodating an arbitrary number of sharpening steps.

MoGe-2 is a feed-forward monocular normal estimator. As shown in Fig. 12(c), the network directly predicts normals from the input image in a single forward pass. Thus, the polarization loss can be backpropagated directly to the image without any intermediate inference stages. This makes Poppy straightforward for feed-forward architectures.

Across all architectures, the backbone networks remain frozen, and Poppy optimizes the learnable offsets before and after the network using polarization-guided losses (Sec. 4.3). The optimization is inserted at architecture-specific points in the inference pipeline, enabling a unified framework compatible with diffusion-based, multi-stage flow-based, and feed-forward normal estimation models.

D Incorporating field-of-view estimation from MoGe

In Tab. 2, we incorporate the field-of-view (FoV), inferred by MoGe-2, into the Poppy polarization guidance optimization. Given the predicted FoV, we compute the focal length as $f = \frac{W}{2\tan(\mathrm{FoV}/2)}$, where W is the image width. Per-pixel viewing directions are then computed by constructing a pinhole camera grid over the image plane. Specifically, pixel coordinates (u, v) are defined with the origin at the principal point (c_x, c_y), with the v-axis pointing upward. The unprojected

Table 2: Quantitative comparison on the PISR dataset. We compare the backbone model (MoGe-2), MoGe-2 with our polarization guidance assuming orthogonal projection (+ Poppy), and our guidance with perspective projection when the field-of-view (FoV) is incorporated (+ Poppy (FoV)). In the FoV variant, the camera FoV is inferred by MoGe-2 and used during optimization. Lower values are better for Mean, Median, and RMSE, while higher values are better for $\text{Acc}_{11.25}$, $\text{Acc}_{22.5}$, and Acc_{30}.

Dataset	Method	Mean	Median	RMSE	$\text{Acc}_{11.25}$	$\text{Acc}_{22.5}$	Acc_{30}
PISR	MoGe-2	12.82	9.82	17.92	0.58	0.87	0.94
	MoGe-2 + Poppy	13.54	11.27	18.00	0.50	0.89	0.95
	MoGe-2 + Poppy (FoV)	**10.87**	**8.05**	**16.02**	**0.69**	**0.92**	**0.96**

ray direction for each pixel is given by

$$\mathbf{d}(u, v) = \left(\frac{c_x - u}{f_x}, \ \frac{c_y - v}{f_y}, \ 1 \right), \tag{4}$$

where $f_x = f$ and $f_y = f \cdot \frac{H}{W}$ are the horizontal and vertical focal lengths, respectively. Each direction is then normalized to obtain the unit viewing direction $\hat{\mathbf{d}}(u, v) = \mathbf{d}(u, v)/\|\mathbf{d}(u, v)\|_2$. Now we compute the elevation angle of a surface with the viewing directions in normals to Stokes conversion, as described in Sec. 3.2.

For the PISR dataset, incorporating FoV significantly improves performance, reducing the angular errors and increasing accuracy across all angular thresholds. This improvement indicates that modeling the camera FoV is important when perspective distortions are present, since the viewing directions used in the polarization model depend on the correct camera geometry.

In scenes with noticeable perspective distortion, assuming an orthographic viewing model can introduce errors in the estimated surface normals. By explicitly incorporating the FoV, the optimization can better account for the true viewing rays and therefore produce more accurate polarization-consistent normals. This result suggests that FoV modeling becomes particularly beneficial when objects are close to the camera and perspective effects cannot be neglected.

E Effect of varying refractive index

To evaluate the sensitivity of Poppy to refractive index, we run additional experiments using $\eta = 3.2$, which is the value adopted in the original SfPUEL [39] work, while our default assumption is $\eta = 1.5$ corresponding to common materials. As shown in Tab. 3, the performance difference between the two settings is negligible across all metrics and scenes, with variations appearing only at the third decimal place in most cases.

Polarization-based normal estimation relates the degree of linear polarization (DoLP) to the elevation angle through Fresnel reflection models as described in Sec. 3.2. Both diffuse and specular DoLP depend on η through rational functions

Table 3: Comparison of different refractive index η settings on the SfPUEL dataset, which contains both dielectric and metallic objects. Poppy assumes a refractive index $\eta = 1.5$, while the SfPUEL paper uses $\eta = 3.2$. To analyze sensitivity to the refractive index assumption, we additionally evaluate Poppy with $\eta = 3.2$. The best results are highlighted in bold.

	Real						Synthetic					
Method	Mean	Median	RMSE	Acc11.25	Acc22.5	Acc30	Mean	Median	RMSE	Acc11.25	Acc22.5	Acc30
MoGe-2	10.84	9.87	12.85	0.60	0.94	0.98	10.93	8.84	13.98	0.62	0.91	0.95
MoGe-2 + Poppy ($\eta = 1.5$)	**8.74**	**7.61**	**10.91**	**0.75**	**0.97**	**0.99**	**9.19**	**6.84**	**12.50**	**0.75**	**0.94**	**0.96**
MoGe-2 + Poppy ($\eta = 3.2$)	8.75	7.62	10.92	**0.75**	**0.97**	**0.99**	**9.19**	**6.84**	**12.50**	**0.75**	**0.94**	**0.96**

whose variation with respect to η is relatively smooth. In the range of refractive indices typically encountered in real materials, the change in ρ_d and ρ_s induced by modifying η mainly results in small shifts in the DoLP.

This explains why using $\eta = 1.5$ (as used in Poppy) or $\eta = 3.2$ (as used in the SfPUEL dataset) yields nearly identical optimization behavior and final normal estimates in our experiments.

F Comparison of Poppy and direct finetuning

Poppy introduces a lightweight test-time guidance approach, which involves just learning per-pixel offsets to the input and output of the backbone network with the backbone weights as frozen. Here, we compare our approach with the alternative of fine-tuning the weights of the backbone estimators, without any learnable offset, using the Stokes loss of the given scene at test-time.

Table 4: Memory usage comparison between Poppy and direct backbone finetuning. Peak memory usage of Poppy and direct finetuning of model weights are reported for each backbone. Direct finetuning requires gradients and optimizer states for all backbone parameters, making it infeasible for large models such as Lotus-v2 on a 96 GB GPU.

Backbone	Poppy (GB)	Direct Finetuning (GB)
Marigold	35.12	47.9
Lotus-v2	66.64	>96 (OOM)
MoGe-2	15.33	21.0

Memory comparison of direct backbone finetuning. We compare two settings: our Poppy optimization and direct backbone weights finetuning with the same polarization guidance as Poppy in Tab. 4. During inference, the backbone performs a forward pass, and only the model weights and forward activations required for computation must be stored in memory.

Direct finetuning uses more memory demanding because gradients must be computed for all backbone parameters. This requires storing intermediate activations for backpropagation and allocating additional gradient tensors for every model parameter. As a result, the memory footprint of finetuning is larger than that of Poppy, especially for large models. In particular, finetuning for Lotus-v2 is not possible because the model could not be trained under the available memory budget (96 GB).

Poppy avoids this issue by keeping the backbone weights frozen and optimizing only a small set of offset variables (image offsets, normal offsets, and radiance maps). Since gradients are not computed for the backbone parameters, the backbone can be executed in inference mode while the optimization updates only the offset tensors. Consequently, the memory overhead of Poppy is less than direct finetuning.

Table 5: Comparison of Poppy and finetuning. We compare the backbone models (None), direct backbone finetuning with the polarization loss (Finetune), and our polarization-guided optimization (Poppy). The best result for each backbone and dataset is highlighted in bold. Finetuning results for Lotus-v2 are not reported because the model could not be trained due to GPU memory constraints.

Backbone	Variant	Synthetic				Real		
		SfPUEL	NeRSP	NeISF	DeepPol	SfPUEL	NeRSP	PISR
Marigold	None	15.19	21.47	20.03	27.29	17.45	19.03	18.06
	Finetune	13.11	17.15	12.38	21.38	**11.02**	17.23	17.44
	Poppy	**12.45**	**16.55**	**12.33**	**21.07**	12.12	**17.01**	**16.71**
Lotus-v2	None	13.14	17.85	15.30	19.80	12.38	17.76	13.90
	Poppy	**10.22**	**13.82**	**9.92**	**15.08**	**9.64**	**15.48**	**12.82**
MoGe-2	None	10.93	16.08	13.55	15.95	10.84	15.63	**12.82**
	Finetune	9.53	12.32	**8.64**	**12.63**	**8.34**	15.06	15.30
	Poppy	**9.19**	**11.98**	9.58	12.80	8.74	**14.51**	13.54

Quantitative comparison with direct finetuning. Tab. 5 compares the backbone models, direct backbone finetuning with the polarization guidance, and Poppy. The best learning rate is searched and used for each backbone (10^{-6} for Marigold and 10^{-7} for MoGe-2) for comparison. Overall, both finetuning and Poppy achieve similar performance across datasets, with only small differences in MAE. Fig. 13 shows that the similar performance is not due to insufficient finetuning steps but rather that both methods converge to comparable solutions. This indicates that most of the benefits of polarization-guided adaptation can be obtained without updating the backbone weights.

Fig. 13 further illustrates the optimization dynamics of both approaches. While finetuning converges, its loss trajectory is noticeably less stable and oscillates. In contrast, Poppy converges smoothly, reflecting the well-conditioned nature of optimizing small offset variables against a frozen backbone.

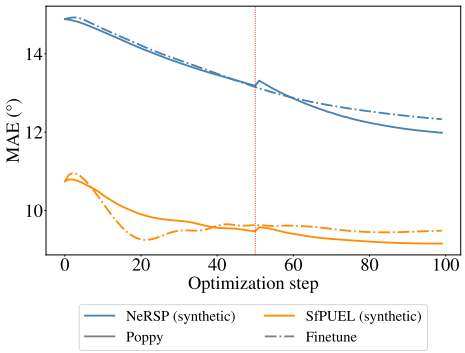


Fig. 13: MAE convergence curves over optimization steps for Poppy and direct backbone weights finetuning (MoGe-2) across SfPUEL and NeRSP synthetic datasets. Poppy converges more smoothly, while finetuning exhibits unstable oscillations despite very small learning rates.

Importantly, Poppy incurs less computational overhead while achieving similar performance to the finetuning approach. As a result, it is more memory efficient and practical for various models.

G Comparison of learned diffuse–specular radiance decomposition

Fig. 14 compares the predicted diffuse and specular radiance components with the corresponding ground-truth decomposition in the NeISF dataset. The predicted L_d closely reproduces the ground-truth diffuse appearance, preserving the global shading patterns of the objects, while L_s effectively isolates reflective regions, particularly near the object boundaries and corners where strong specular reflections from surrounding surfaces occur. The absolute error maps confirm low reconstruction error across most valid pixels, and the PSNR and SSIM values further confirm that the decomposition is quantitatively consistent with the ground truth decomposition.

H Image editing from our learned decomposition

Fig. 15 demonstrates appearance editing applications, mentioned in Sec. 5.3, enabled by the diffuse and specular radiance components learned using Poppy. Edits applied to the diffuse component L_d modify the surface color appearance while preserving the original specular highlights and reflection patterns, as shown in the recoloring example. In contrast, edits to the specular component L_s affect the material's reflective properties. Changing the hue of L_s produces a metallic appearance, simulating materials with strong, colored specular reflections across the entire surface. By independently controlling L_d and L_s, realistic material modifications can be achieved while maintaining consistent lighting and shading.

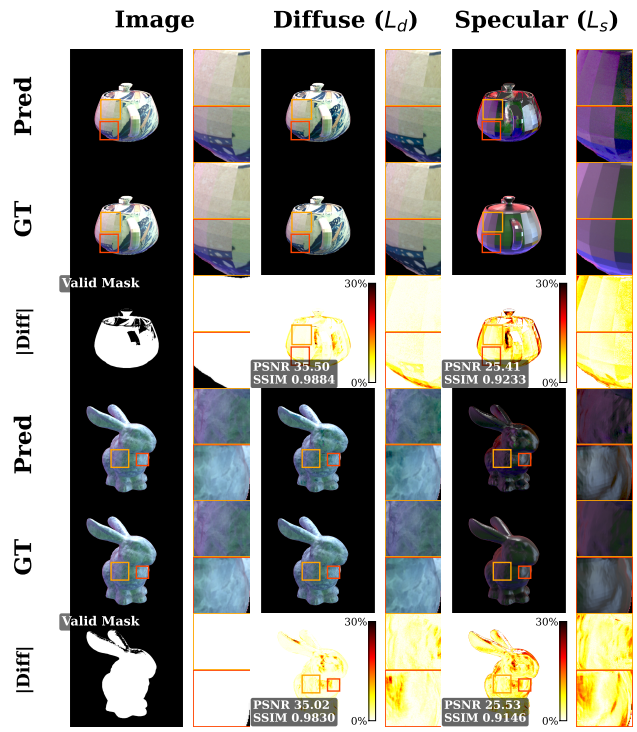


Fig. 14: Qualitative comparison of the predicted diffuse (L_d) and specular (L_s) radiance components with ground truth (GT). The absolute error maps (shown as percentages) visualize per-pixel reconstruction error within the valid polarization mask, with overall PSNR and SSIM values reported as labels on each error map. The valid mask indicates pixels where polarization-based constraints are reliable and used during optimization. The results show that the predicted decomposition closely matches the ground truth across both diffuse and specular regions, with low absolute errors over most valid pixels.

Fig. 15: Applications of radiance decomposition from Poppy for appearance editing. Given the estimated diffuse (L_d) and specular (L_s) components, different visual effects can be achieved by manipulating each component independently. Recoloring modifies the diffuse radiance L_d to change the surface color while preserving specular reflections, while the metallic appearance effect is produced by editing the specular radiance L_s. These examples demonstrate that the decomposition enables flexible and physically interpretable image editing.