EUROGRAPHICS 2026 / B. Masia and J. Thies
(Guest Editors)

COMPUTER GRAPHICS *forum*
Volume 45 (*2026*), *Number 2*

Neural Local Inter-reflection Modeling for Garment Fold Rendering: Supplementary Material

Jooeun Son[1] Nuri Ryu[1] Gyoonseo Kim[1] Joo Ho Lee[2] Seungyong Lee[1]

[1]POSTECH, South Korea
[2]Sogang University, South Korea

Table 1: *Important symbols for the BLIDF derivation.*

Notation	Definition
$\mathcal{N}(x_\mathrm{o})$	Spherical local neighborhood around point x_o.
r	Radius of the local neighborhood $\mathcal{N}(x_\mathrm{o})$.
x_o	Evaluation point; center of local neighborhood $\mathcal{N}(x_\mathrm{o})$.
x_i	Incident point within local neighborhood $\mathcal{N}(x_\mathrm{o})$.
dA_i	Differential area element at an incident point x_i.
ω_i	Incident light direction.
ω_o	Outgoing (view) direction.
Ω	Hemisphere of integration for incident directions.
$\lvert\cos\theta_i\rvert$	Foreshortening term for incident light.
$L_\mathrm{o}^\mathrm{local}$	Local outgoing radiance at x_o.
L_i	Incident radiance.
$V(x_\mathrm{o},\omega_\mathrm{i})$	Direct visibility from point to emitter direction ω_i
$\bar{V}_\mathcal{N}(x_\mathrm{o},\omega_\mathrm{i})$	Average visibility of the neighborhood region $\mathcal{N}(x_\mathrm{o})$.
S_base	Base micro-scale BRDF (single-bounce component).
S_surf	Surface scattering transfer function within $\mathcal{N}(x_\mathrm{o})$.
$\bar{S}$	Bidirectional Local Inter-reflection Distribution Function
S_θ	Neural approximation of $\bar{S}$ parameterized by θ.

Table 2: *Descriptors and Optimization Parameters*

Symbol	Dims.	Definition
Descriptors		
ϕ_S	9	Local fold geometry descriptor at x_o.
ϕ_hv	1	Hill-Valley component of ϕ_S, encoding local concavity.
ϕ_poly	8	Polynomial Surface component of ϕ_S, encoding directional variation.
ϕ_L	2	Incident illumination descriptor encoding incident direction ω_i at both local and regional scale.
ϕ_M	1	Material descriptor encoding diffuse albedo.
Optimization Parameters		
Symbol	Dims.	Definition
θ	1.4K	BLIDF Network Parameters
η_{ASG}	6×2	ASG parameters, with 6 parameters per lobe. ($N=2$)

1. Symbol Tables

In Table 1, we summarize the core mathematical notations for the BLIDF derivation. In Table 2, we detail the input features used to condition the neural networks and the parameters updated during the training process.

2. Details on Modeling Neighborhood Visibility

In this section, we provide details on approximating the neighborhood visibility term $\bar{V}_\mathcal{N}$: using (i) an Anisotropic Spherical Gaussian-based (ASG) method, where ASG parameters are jointly learned with the network, and (ii) a sampling-based heuristic, which does not require any pre-training and is directly applicable to unseen garments.

2.1. ASG-based Visibility

To effectively handle the high complexity of the neighborhood visibility function, we compute the visibility function $\bar{V}_\mathcal{N}$ as the maximum between two terms: (i) direct visibility $V(x_\mathrm{o},\omega_\mathrm{i})\in\{0,1\}$, which is computed with standard ray-occlusion testing, and (ii) the residual visibility term $\bar{V}_{\mathcal{N},res}$:

$$\bar{V}_\mathcal{N}(x_\mathrm{o},\omega_\mathrm{i}) = \max(V(x_\mathrm{o},\omega_\mathrm{i}), \bar{V}_{\mathcal{N},res}(x_\mathrm{o},\omega_\mathrm{i})). \tag{1}$$

The direct visibility term handles high-frequency details (e.g., sharp shadows) from global occluders, allowing the residual term to focus only on the low-frequency components. In this formulation, we model the residual visibility term with ASGs.

Anisotropic Spherical Gaussians (ASG) [XSD*13] have been adopted in previous work for modeling meso-scale shadowing of plies and yarns in woven fabric [ZJA*23]. While folds are larger in scale than plies and yarns, folds exhibit a similar elongated geometry, resulting in a highly directional visibility distribution. Leveraging this structural similarity, we model neighborhood visibility as a sum of N ASG lobes:

$$\begin{aligned}\bar{V}_{\mathcal{N},res}(x_{\mathrm{o}},\omega_{\mathrm{i}}) &= \sum_{l=1}^{N} ASG_l(\omega_{\mathrm{i}};[\mathbf{x},\mathbf{y},\mathbf{z}],[\sigma_x,\sigma_y],C) \\ &= \sum_{l=1}^{N} C\cdot\max(\mathbf{v}\cdot\mathbf{z},0)\cdot\exp\left(-\sigma_x(\mathbf{v}\cdot\mathbf{x})^2-\sigma_y(\mathbf{v}\cdot\mathbf{y})^2\right),\end{aligned} \tag{2}$$

where $\mathbf{z},\mathbf{x},\mathbf{y}$ are the lobe, tangent and bi-tangent axes, respectively, and σ_x,σ_y are the bandwidths for the $\mathbf{x}$- and $\mathbf{y}$- axes, respectively. C is the lobe amplitude. Each surface point has 6 parameters for each ASG lobe: the lobe vector $\mathbf{z}\in\mathbb{R}^3$, bandwidths $\sigma_x,\sigma_y\in\mathbb{R}^2$ and amplitude $C\in\mathbb{R}$. These parameters are jointly optimized with the networks. In practice, we use 2 ASG lobes ($N=2$).

2.2. Sampling-based Heuristic

The sampling-based heuristic is needed to correctly approximate neighborhood visibility for unseen garments, without any pre-training. Using this heuristic, neighborhood visibility is approximated as the ratio of neighboring surface samples of x_{o} that are unoccluded in direction ω_{i}. We first sample N_s surface points x_i inside $\mathcal{N}(x_{\mathrm{o}})$ by casting probe rays towards the surface, similar to BSSRDF importance sampling methods [KKCF13]. Afterwards, we perform ray-occlusion tests for each sample in direction ω_{i}, and calculate the ratio of unoccluded samples:

$$\bar{V}_{\mathcal{N}}(x_{\mathrm{o}},\omega_{\mathrm{i}}) = \frac{1}{N_s}\sum_{j=1}^{N_s} V(x_{i,j},\omega_{\mathrm{i}}). \tag{3}$$

In practice, we use 4 surface samples ($N_s=4$).

3. Additional Experiments

3.1. Analysis on Network Architecture

In Table 3 and Fig. 1, we evaluate our architectural design choices and the impact of Spherical Harmonics (SH) order on local inter-reflection modeling. We evaluate our factorized, SH-based approach against two primary architectural baselines: (i) an *End-to-End* design, which utilizes a single network to predict SH coefficients without separating intensity and directional distribution, and (ii) a *Per-Direction* design, which bypasses SH representations by taking a query direction vector as input to directly estimate radiance.

We created a benchmark dataset of 8 garment items rendered under diverse lighting, camera, and material configurations ($\sim$ 200 images per garment). We train each baseline for a single garment, and render all images in the dataset. We report the "Seen RMSE" evaluated on the garment used for training and the "Unseen RMSE" evaluated across the remaining 7 garments to assess generalizability.

Architecture factorization. Our results demonstrate that the factorized architecture is critical for generalization. In Table 3, while the End-to-End baseline (Row 1) utilizes the same SH order as our model (Row 2), it fails to generalize to unseen garments, yielding a significantly higher RMSE of 0.03398 compared to our 0.001915 for unseen garments. This disparity is visually evident in Fig. 1: while the End-to-End method performs adequately on the training set ("Seen Garment"), it fails to capture intensity variances in the concave and convex regions of the "Unseen Garment". This confirms that explicitly separating the inter-reflection process into intensity and directional distribution sub-networks is essential for learning a generalizable model.

SH rationale. We observe that the choice of SH order presents a trade-off between training accuracy and generalizability. While lower-order expansions (Rows 3 & 4) provide slightly better accuracy for seen data, SH order 3 is required for robust generalization across unseen garments. This trend is evident in Fig. 1, where lower order results (Columns 2 & 3) exhibit intensity overestimation in concave regions for the "Unseen Garment". Notably, the Per-Direction design (Row 5) yields a marginally higher error for unseen garments compared to our SH-based model. While the difference is small, it requires a significantly larger parameter count (1.8K vs. our 1.4K) to achieve similar rendering accuracy. This suggests that the SH-based formulation provides a more effective inductive bias for local light transport, achieving high fidelity with parameter efficiency.

3.2. Analysis on Shape Descriptors

In this section, we analyze the effectiveness of our proposed descriptor ϕ_{S}, which combines hill-valley and polynomial surface components. We conduct two experiments: first, we ablate the components of our descriptor to justify their combination; second, we compare our geometric features against alternative descriptors, including standard neural encoding methods.

Component ablation. We evaluate the necessity of combining the hill-valley ϕ_{hv} and polynomial descriptors ϕ_{poly} on the same benchmark dataset used in Section 3.1. As shown in Table 4 and Fig. 2, using either component in isolation results in higher RMSE for both seen and unseen garments. Specifically, the "Full" descriptor achieves an unseen RMSE of 0.001915, whereas using only hill-valley or polynomial features increases the error to 0.001961 and 0.002113, respectively. This confirms that while ϕ_{hv} captures essential concavity, ϕ_{poly} provides the necessary structural detail for accurately modeling the geometric features of garment folds.

Comparison with alternative descriptors. In Fig. 3, we provide a visual comparison between our geometric descriptor ϕ_{S} and two common alternatives: curvature and absolute 3D position with sinusoidal encoding. For curvature, we use principal values computed via scale-dependent quadratic fitting using Meshlab [CMRC]. Our descriptor has 9 dimensions, compared to 2 for curvature and 12 for absolute positional encoding. Despite its higher dimensionality, the positional encoding tends to oversmooth sharp fold details, failing to capture the sharp intensity variation between the geometric peaks and valleys of the skirt. Conversely, curvature-based inputs can introduce artifacts at geometric boundaries like skirt hems. By effectively capturing the salient geometric information relevant to local inter-reflections, our descriptor achieves the lowest RMSE, demonstrating its efficacy for high-fidelity garment rendering.

Comparison with neural encodings. We further compare our geometric descriptor against positional inputs processed with various neural encodings methods, including hash grid-based [MESK22], permutohedral [RB23], and sinusoidal encodings. Because the en-

Table 3: *Ablation Study of Network Architecture and SH Order. We evaluate our factorized architecture against two baselines ("End-to-End" and "Per-Dir") on our benchmark dataset. The factorized structure with SH Order 3 provides the best generalization to unseen garments with fewer parameters than other baselines.*

Architecture	Representation	Seen RMSE ↓	Unseen RMSE ↓	Time (s) ↓	Params
End-to-End	SH (Ord 3)	0.010355	0.033398	**0.0988**	2.0K
Factorized	**SH (Ord 3, Ours)**	0.009779	**0.001915**	0.0993	1.4K
Factorized	SH (Ord 2)	0.009739	0.002451	0.0989	1.4K
Factorized	SH (Ord 1)	**0.009720**	0.002491	0.0995	1.3K
Factorized	Per-Direction	0.009857	0.002000	0.1055	1.8K

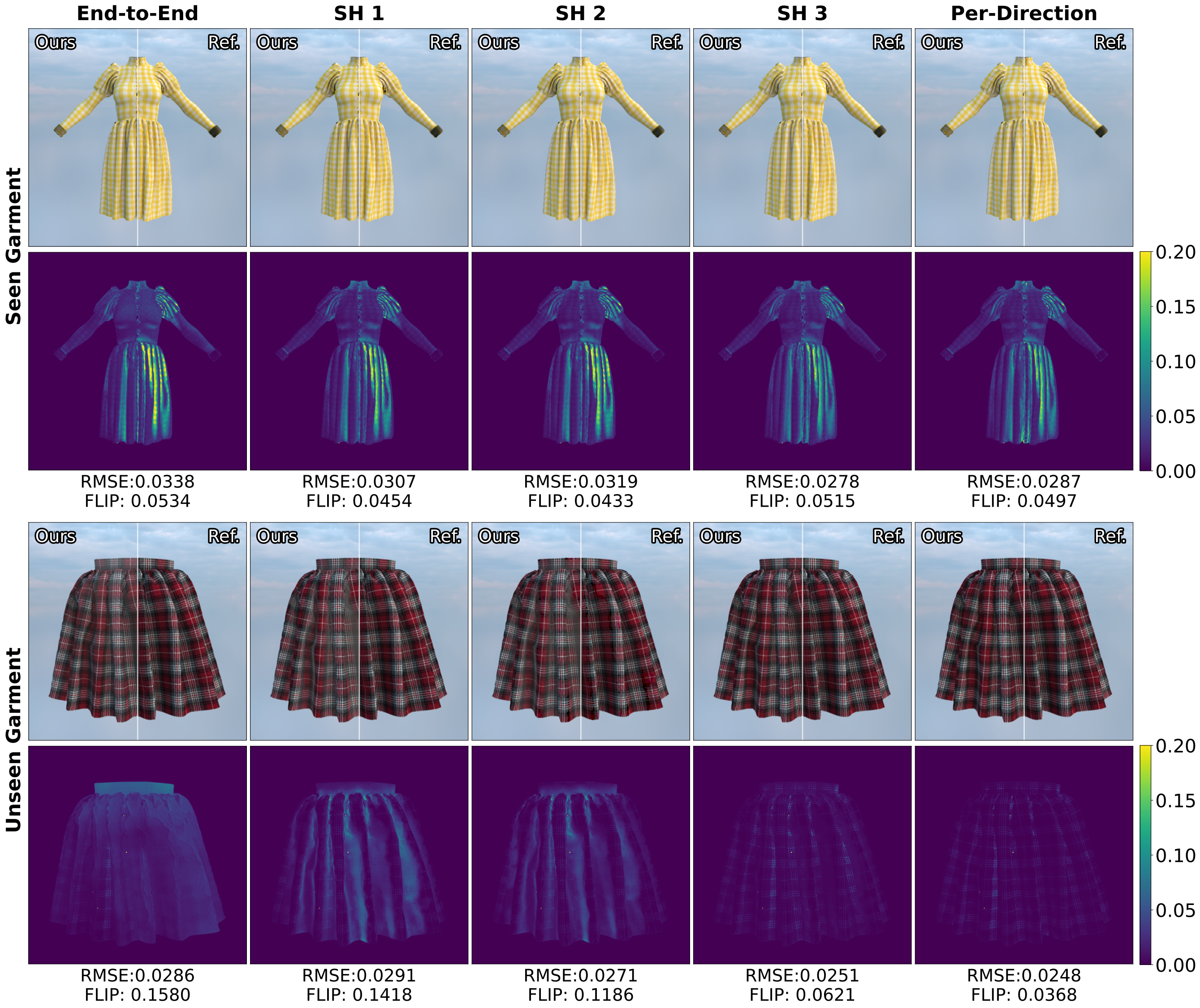


Figure 1: *Analysis on network architecture. We evaluate the rendering quality of various architecture models across seen and unseen geometries. In the "Seen Garment" experiment (yellow dress), the models are tested on the same garment used for training. In the "Unseen Garment" experiment (skirt), we evaluate generalization by testing the models on a completely new mesh topology. In the case of the "Seen Garment" scene, RMSE and FLIP errors are minor for all models. However, the "End-to-End" and lower-order SH configurations (Order 1 and 2) exhibit a wider error gap when applied to "Unseen Garment".*

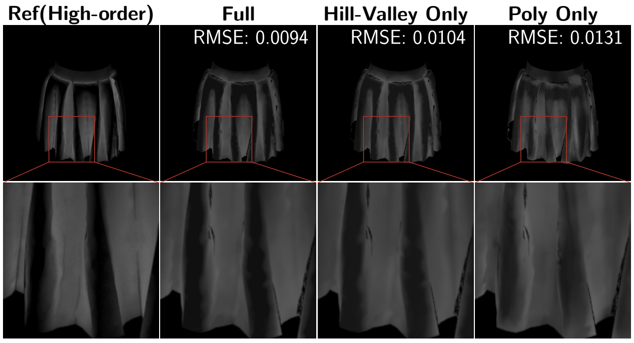


Figure 2: *Ablation on shape descriptor. We report the RMSEs of different variations of the shape descriptor. Our combination of the hill-valley and polynomial descriptors yields the lowest error, as can be seen in the clear differences between hills and valleys.*

Table 4: *Descriptor Component Ablation. We compare our full descriptor against individual hill-valley and polynomial baselines. The results demonstrate that using the combination of the hill-valley and polynomial descriptors is necessary for high-fidelity rendering.*

Metric	Full (Ours)	Hill-Valley Only	Poly Only
Seen RMSE ↓	**0.009779**	0.009965	0.009886
Unseen RMSE ↓	**0.001915**	0.001961	0.002113
Time (s) ↓	0.0993	0.0928	0.0983
Param(#)	1.4K	1.3K	1.4K

coding methods take absolute 3D positions as input rather than local geometric information, the learned model is scene-specific and fails to generalize to unseen garment geometries. Thus, for this experiment, we only evaluate the performance on the garment used during training ("Seen"). As detailed in Table 5, our proposed geometric descriptor yields the lowest error (0.001655 RMSE), outperforming hash grid-based (0.002217) and permutohedral (0.001773) encodings. This superiority holds even when increasing the number of encoding parameters, or using UV coordinates instead of 3D positions (Row 5). Our descriptors effectively reduce the dimensionality of the problem by extracting salient local geometric information, granting superior generalization and rendering accuracy.

3.3. Analysis on Visibility Method

We conduct a quantitative comparison of our two visibility solutions, learned ASGs and the sampling-based heuristic, to validate their efficacy across different use cases. For each experiment, the BLIDF and ASG parameters are jointly optimized during training. At render time, we evaluate the impact of switching to different visibility configurations. As shown in Table 6, RMSE remains consistent between the learned ASG (Rows 1-2) and the sampling-based heuristic (Rows 3-4). Crucially, the "Sampling (Unseen)" test (Row 5), where a BLIDF pre-trained on a different garment is used for rendering, maintains an error level (0.05907) similar to the "Seen" cases. This confirms the robust generalizability of our model when paired with a training-free visibility solution, offering a practical path for rendering unseen garments without the need for per-scene optimization.

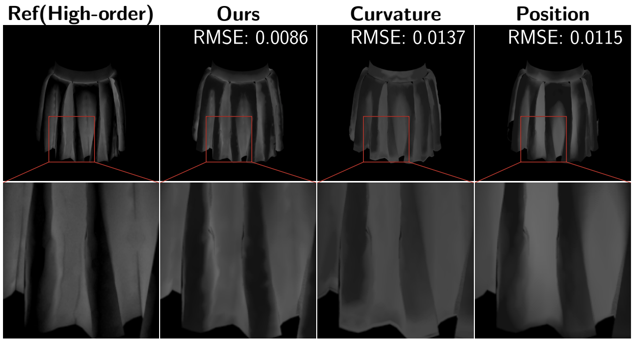


Figure 3: *Comparison with alternative descriptors. Curvature descriptors (Column 3) introduce artifacts at mesh boundaries (e.g., skit hems), while absolute positional input (Column 4) tends to overly smoothen sharp fold details. Our proposed descriptor (Column 2) achieves the lowest RMSE and best preserves salient features by effectively capturing relevant geometric information.*

3.4. Analysis on Fold Geometry

In Fig. 5, we analyze the effectiveness of our method across diverse fold shapes. We procedurally generated planar surfaces with various fold patterns: sinusoidal waves of increasing amplitude, and non-sinusoidal horseshoe shapes with different opening widths.

In the directional emitter setting (Rows 1 & 2), our method produces smooth, high-quality results with only 1 sample per pixel (SPP), outperforming LocalPT (4 SPP) in terms of RMSE and rendering time.

In the environmental lighting setting, our method outperforms both LocalPT and PT in terms of rendering quality and time for all datasets. Comparison with LocalPT (Rows 3 & 4) demonstrates that our method has a higher gain in rendering quality compared to PT (Rows 5 & 6), indicating that our method is especially suitable for scenes dominated by local inter-reflections.

For the sinusoidal waves, increasing the amplitude leads to increased rendering time for path tracers, due to the increased number of bounces within folds. In our experiments, both PT and LocalPT demonstrate roughly equal rendering quality across different shapes. On the other hand, our method has relatively higher error for shapes with deeper valleys ("Sine-High", "Wave").

3.5. Additional Experimental Results

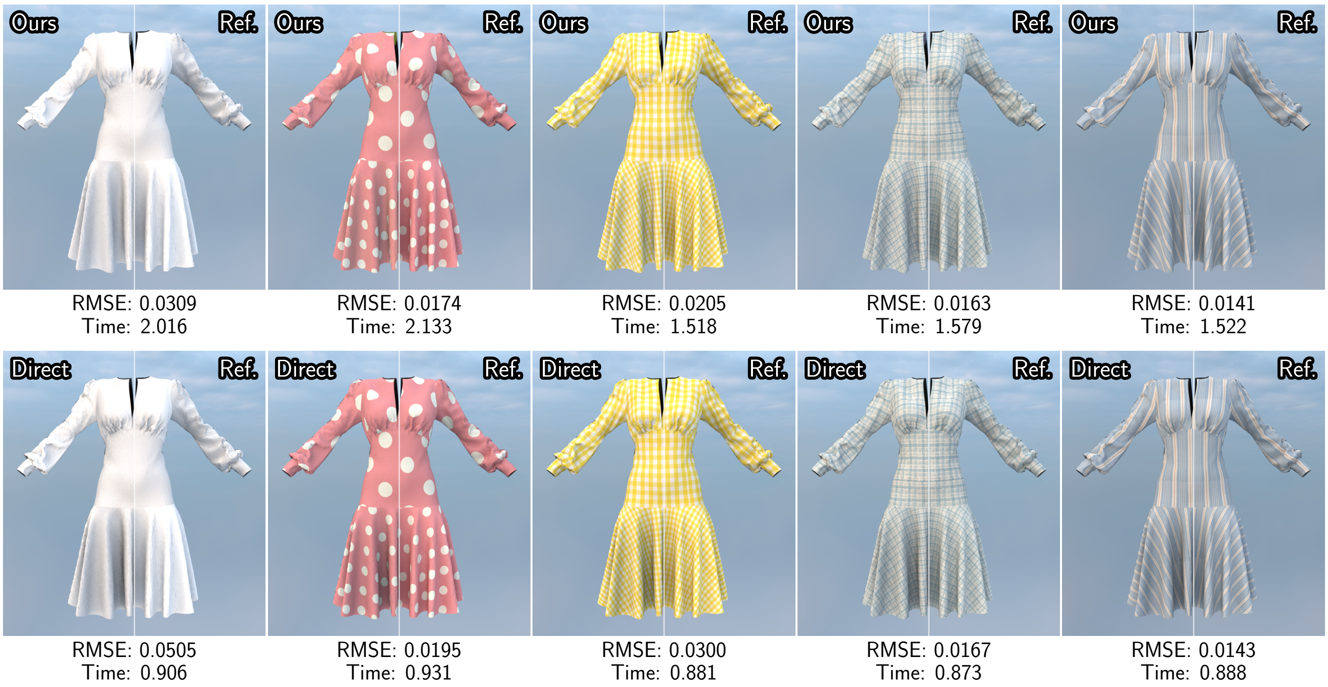
Textured garments. In Fig. 4, we provide additional results on textured garments. Compared to the direct integrator renderings, our method yields more realistic results, especially in the dense folds of the skirt.

Table 5: *Comparison with neural encodings. We compare our geometric descriptor against positional inputs with various neural encodings. The results demonstrate that our geometric descriptor outperforms all position-based encoding methods in terms of rendering accuracy, rendering speed, and parameter count.*

Method	Seen RMSE ↓	Time (s) ↓	Net. Params	Enc. Params
Ours	**0.001655**	**0.0983**	**1.4K**	—
HashGrid	0.002217	0.1096	1.5K	71K
Permutohedral	0.001773	0.1076	1.5K	71K
Sine	0.001967	0.0994	1.9K	—
Sine (UV)	0.002142	0.1007	2.0K	—

Table 6: *Visibility method comparison. We compare our learned ASG-based visibility against the sampling-based heuristic. The results demonstrate that the heuristic provides a robust, training-free alternative for unseen geometries with comparable RMSE.*

Method	Configuration	Seen/Unseen	RMSE ↓	Time (s) ↓
ASG (Learned)	1 Lobe	Seen	0.05947	0.54750
ASG (Learned)	2 Lobes	Seen	0.05940	0.56837
Sampling	2 Rays	Seen	0.05853	0.77263
Sampling	4 Rays	Seen	**0.05807**	1.06660
Sampling	4 Rays	Unseen	0.05907	1.17827



RMSE: 0.0309 Time: 2.016

RMSE: 0.0174 Time: 2.133

RMSE: 0.0205 Time: 1.518

RMSE: 0.0163 Time: 1.579

RMSE: 0.0141 Time: 1.522

RMSE: 0.0505 Time: 0.906

RMSE: 0.0195 Time: 0.931

RMSE: 0.0300 Time: 0.881

RMSE: 0.0167 Time: 0.873

RMSE: 0.0143 Time: 0.888

Figure 4: *Textured garments. Our method yields more realistic results compared to the direct integrator, especially in the dense folds.*

4. Implementation Details

4.1. Computing Shape Descriptors

Polynomial surface descriptor ϕ_{poly}. Following the approach of Vicini et al. [VKJ19], we extract second-order polynomial coefficients for each vertex. The algorithm samples surface points within a spherical neighborhood $\mathcal{N}(x_o)$ and employs a weighted least-squares optimization [SOS04] to solve for the coefficients. While Vicini et al. [VKJ19] determine the neighborhood size based on material scattering parameters, our neighborhood radius r is fixed to encapsulate a single fold geometrically. As specified in our training configuration, we set $r = 0.15$ for skirts and $r = 0.10$ for dresses.

Hill-Valley descriptor ϕ_{hv}. Hill-Valley descriptor ϕ_{hv} is a scalar representing local concavity. We iteratively determine the maximum radius r_{sphere} of a virtual sphere tangent to x_{o} that does not intersect the local geometry. Utilizing the neighborhood samples $P = \{p_i\}_{i=1}^{n}$ from the polynomial fitting step, we solve the following optimization problem using the SLSQP algorithm [Kra88]:

$$L = \sum_{p_i \in P} w_i(||p_i - c|| - r_{sphere})^2, \tag{4}$$

where $c = x_{\mathrm{o}} + r_{sphere}\,\vec{n}$ is the sphere center. The weight w_i is defined as:

$$w_i = \begin{cases} \lambda & \text{if } \|p_i - c\| < r_{sphere} \quad \text{(Self-intersection)} \\ 1 & \text{otherwise} \quad \text{(Surface fitting)} \end{cases}$$

This weighting scheme penalizes samples inside the sphere to prevent self-intersections ($\lambda = 1000$), while encouraging the sphere boundary to align with the surface samples. We perform this optimization until convergence with the maximum of 100 iterations. In a planar or convex region, where a tangent sphere of finite radius is ill-defined, the optimization fails to converge. Then, we clamp r_{sphere} to the neighborhood radius r. To handle the two-sidedness of garments, we repeat the algorithm for the back side using the flipped normal $-\vec{n}$, instead of $\vec{n}$, for computing the sphere center while reusing the same surface samples for the fitting process.

Neighborhood normal $\vec{n}_{\mathcal{N}}$. The neighborhood normal $\vec{n}_{\mathcal{N}}$ captures the regional orientation of a fold. It is computed by averaging the surface normals of the samples acquired during the polynomial fitting process. As a result of averaging, $\vec{n}_{\mathcal{N}}$ varies smoothly across regions with high-frequency surface details, providing a macroscopic geometric context that complements the local shading normal $\vec{n}$ when the illumination descriptor ϕ_{L} is computed.

4.2. Ground Truth Generation and Baselines

Ground truth images for training and comparison are rendered using LocalPT with a maximum path depth of 16 and a Russian Roulette (RR) depth of 5. For all path tracer comparisons, the baseline path tracers also use a RR depth of 5 to ensure fairness.

References

[CMRC] Cignoni, Paolo, Muntoni, Alessandro, Ranzuglia, Guido, and Callieri, Marco. *MeshLab*. doi: 10.5281/zenodo.5114037 2.

[KKCF13] King, Alan, Kulla, Christopher, Conty, Alejandro, and Fajardo, Marcos. "BSSRDF importance sampling". *ACM SIGGRAPH 2013 Talks*. 2013, 1–1 2.

[Kra88] Kraft, Dieter. "A software package for sequential quadratic programming". *Forschungsbericht- Deutsche Forschungs- und Versuchsanstalt fur Luft- und Raumfahrt* (1988) 6.

[MESK22] Müller, Thomas, Evans, Alex, Schied, Christoph, and Keller, Alexander. "Instant neural graphics primitives with a multiresolution hash encoding". *ACM Transactions on Graphics (TOG)* 41.4 (2022), 1–15 2.

[RB23] Rosu, Radu Alexandru and Behnke, Sven. "PermutoSDF: Fast Multi-View Reconstruction with Implicit Surfaces using Permutohedral Lattices". *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023 2.

[SOS04] Shen, Chen, O'Brien, James F., and Shewchuk, Jonathan R. "Interpolating and approximating implicit surfaces from polygon soup". *ACM SIGGRAPH 2004 Papers*. SIGGRAPH '04. Los Angeles, California: Association for Computing Machinery, 2004, 896–904. isbn: 9781450378239. doi: 10.1145/1186562.1015816. url: https://doi.org/10.1145/1186562.1015816 5.

[VKJ19] Vicini, Delio, Koltun, Vladlen, and Jakob, Wenzel. "A learned shape-adaptive subsurface scattering model". *ACM Transactions on Graphics (TOG)* 38.4 (2019), 1–15 5.

[XSD*13] Xu, Kun, Sun, Wei-Lun, Dong, Zhao, et al. "Anisotropic spherical Gaussians". *ACM Trans. Graph.* 32.6 (Nov. 2013). issn: 0730-0301. doi: 10.1145/2508363.2508386. url: https://doi.org/10.1145/2508363.2508386 1.

[ZJA*23] Zhu, Junqiu, Jarabo, Adrian, Aliaga, Carlos, et al. "A Realistic Surface-based Cloth Rendering Model". *Proceedings - SIGGRAPH 2023 Conference Papers*. Association for Computing Machinery, Inc, July 2023. isbn: 9798400701597. doi: 10.1145/3588432.3591554 1.

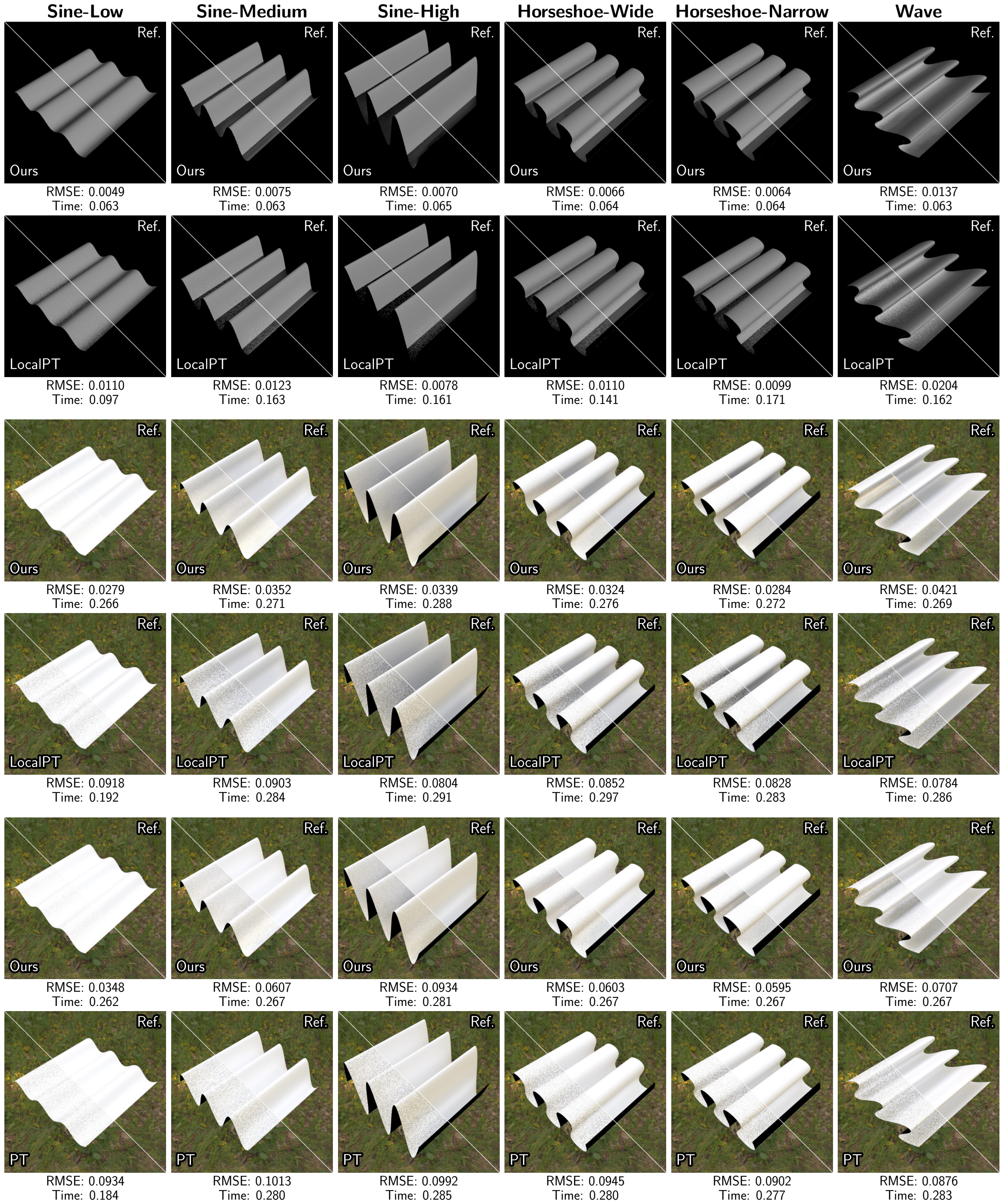


Figure 5: *Fold shape analysis. We visualize equal rendering time results in various settings, comparing our method against LocalPT and PT. Directional Lighting (Rows 1 & 2): Our method, using 1 sample per pixel (SPP), is compared against LocalPT rendered at 4 SPP. Environment Lighting (Rows 3 $\sim$ 6): For Rows 3 & 4, our method (128 SPP) is compared with LocalPT (8 SPP). For Rows 5 & 6, our method (64 SPP) is compared versus PT (8 SPP).*