On Cosine Prior Distributions for Neural Path Guiding

J. Gutsch[1], M. Dereviannykh[1], J. Hanika[1],

[1]Karlsruhe Institute of Technology, Germany

Abstract

Monte Carlo rendering efficiency depends on the quality of importance sampling distributions. Neural path guiding addresses this by learning adaptive distributions using normalizing flows, which transform simple prior distributions into target distributions. We explore how prior distribution choice affects learning efficiency and show that aligning the prior with components of the rendering integral simplifies the learning task, enabling the use of smaller models. Our cosine-distributed prior, matching the cosine-weighted hemisphere term of the rendering equation, achieves faster convergence and lower noise than uniform priors, with particularly strong improvements in scenes with high geometric complexity.

CCS Concepts

• ***Computing methodologies*** → *Ray tracing; Machine learning;*

1. Introduction

Photorealistic rendering involves simulating light transport using the rendering equation [Kaj86]:

$$L(\mathbf{x},\omega_o) = L_e(\mathbf{x},\omega_o) + \int_\Omega I(\mathbf{x},\omega_i)\,\mathrm{d}\omega_i\,, \tag{1a}$$

$$I(\mathbf{x},\omega_i) = f(\omega_o,\mathbf{x},\omega_i)\,L(\mathbf{x},\omega_i)\,\cos\theta_i\,. \tag{1b}$$

Due to the high complexity of rendered scenes and the recursive nature of the integral, Equation 1 cannot be solved analytically but is instead numerically estimated using Monte Carlo integration [Vea98]. The major downside of this method is that it introduces variance which is visible as noise in the final render under a limited sample budget. Common variance reduction techniques like BSDF sampling and emitter sampling work well for some cases but perform poorly in others, as they each only account for a single term of the integrand (Equation 1b) of the rendering equation.

In this paper we implement a neural path guiding technique, introduce a cosine-weighted prior distribution and compare it against two different uniform priors. We show that the cosine prior outperforms the uniform priors, especially in geometrically complex scenes and for few training iterations. Additionally we study how the performance of each prior changes when the model size is reduced and observe the cosine prior achieving better results as well.

2. Previous Work

Path guiding techniques improve rendering efficiency by learning scene-specific importance distributions, often represented through parametric trainable models. Vorba et al. [VKv*14] optimize a Gaussian mixture model using an *Expectation-Maximization* (EM) algorithm to learn the distribution of incident radiance $L(\mathbf{x},\omega_i)$ while Müller et al. [MGN17] learn the same distribution using adaptive data structures. In both previous cases the distribution is dependent on 5D conditional information, namely the world position $\mathbf{x}$ and the incident light direction ω_i. Learning only the incident light distribution is not effective for glossy and specular materials as directions with bright light might not scatter in the outgoing direction. Solving this problem requires learning the product distribution of the BSDF and incident light. To obtain an approximation for the product distribution Herholz et al. [HEV*16] learns separate Gaussian mixtures for the BSDF and the incident light field and combines them analytically. Recently multiple works have used neural methods: For example, [DWL23] and [HIT*24] utilize neural networks to predict parametric distributions. The authors of [FHK25] factorize the distribution into marginal and conditional distributions, discretize them, and generate samples directly.

Neural importance sampling (NIS) [MMR*19] directly targets the distribution defined by the rendering integrand $I(\mathbf{x},\omega_i)$ (see Equation 1b). For a shading point $\mathbf{x}$ the desired sampling density over incident directions is

$$p(\omega_i \mid \mathbf{x}) \;\propto\; I(\mathbf{x},\omega_i),$$

and NIS constructs an invertible mapping $\omega = F_\phi(z;\mathbf{x})$ that transforms samples $z \sim q(z)$ drawn from a simple prior into directions ω. Normalizing flows realize F_ϕ as a composition of alternating coupling layers G_i [DSDB17], where each layer transforms one subset of coordinates while keeping the others fixed. For 2D samples $x = (x_0, x_1)$ and context c, the two types of layers are :

$$G_0(x_0,x_1) = (x_0, g_{\theta(x_0,c)}(x_1))\,, \tag{2a}$$

$$G_1(x_0,x_1) = (g_{\theta(x_1,c)}(x_0), x_1)\,. \tag{2b}$$

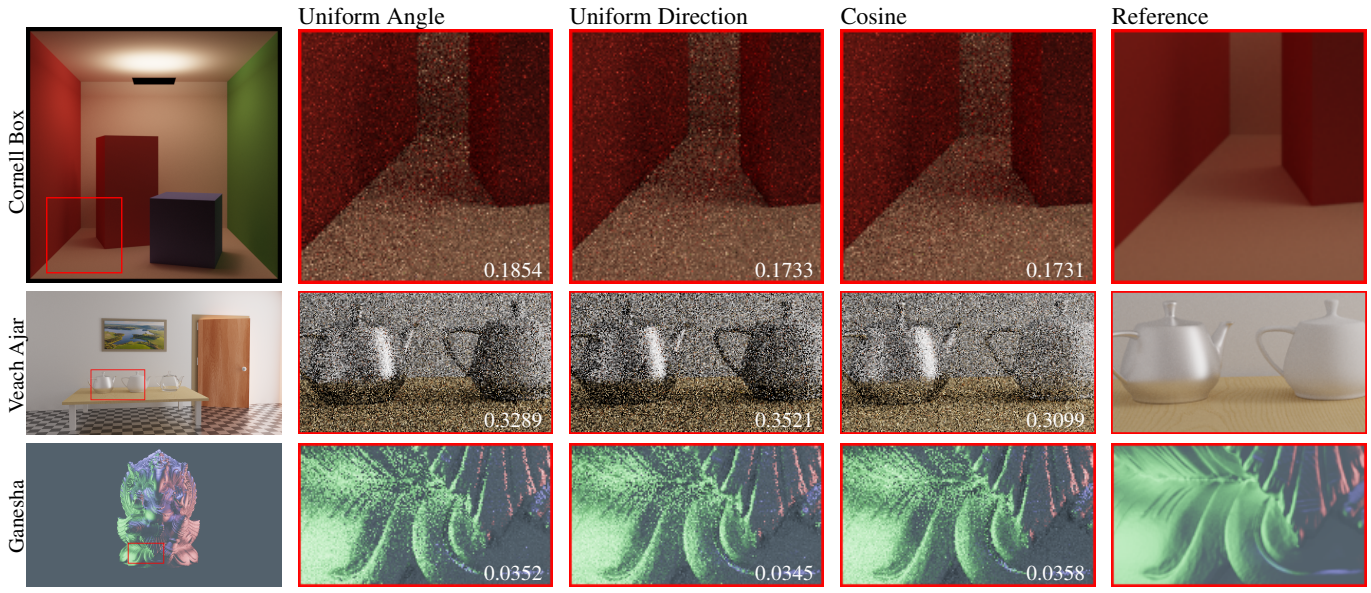

Figure 1: Equal sample (32 spp) comparisons of different priors on multiple scenes. We report the mean average percentage error for each image. The effectiveness of the cosine prior over the uniform depends on the scene, but can be seen especially well on high frequency geometric detail such as on the Ganesha statue

The full mapping is then the composition of L such layers, $F_\phi = G_{L-1} \circ \cdots \circ G_0$. Here g_θ is an invertible mapping, such as a piecewise-polynomial spline, with parameters θ predicted by an MLP conditioned on the static coordinates and the surface context c (position, normal, incoming direction). This structure makes inversion and determinant computation tractable, allowing evaluation of the density via the change-of-variable formula:

$$p(\omega \mid \mathbf{x}) = q(F_\phi^{-1}(\omega;\mathbf{x})) \left|\det J_{F_\phi^{-1}}(\omega;\mathbf{x})\right|. \tag{3}$$

While expressive, flows require multiple layers and per-layer networks, which adds computational cost.

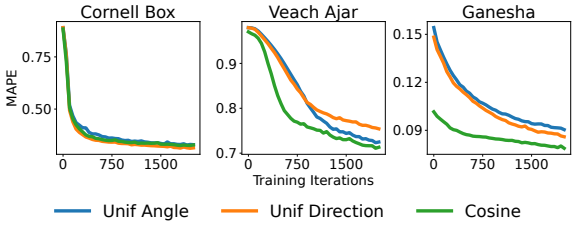

Figure 2: Convergence plots of MAPE during training. Every 50 iterations a full image was rendered with 4 spp to measure the error.

3. Neural path guiding with the Cosine Prior

We implement NIS using rational quadratic splines [DBMP19] instead of the quadratic splines used in the original work [MMR*19], and parameterize world-space directions in spherical coordinates. In our case, the flow is a composition of only two coupling layers ($L = 2$), $F_\phi = G_1 \circ G_0$, with $K = 8$ spline segments and spline parameters predicted by a small MLP with one hidden layer of 16 neurons. Sampling from our model requires drawing from the chosen prior and performing two MLP evaluations to build the transforms G_0 and G_1 (see Equation 2). We encode the features as follows: positions and the static coordinates are encoded with one-blob encoding [MMR*19] and directions with spherical harmonics (first 4 levels).

A common choice of prior distributions when using normalizing flows for general problems is a Gaussian because it covers an unbounded domain. The domain of the world space directions encoded in spherical coordinates is not unbounded however, so different priors can be used like the uniform prior used in NIS. A major downside of the uniform prior is that for non-transmissive materials the density must always be moved away from the regions corresponding to the lower hemisphere, because the object shadows itself in those directions. We propose to condition the prior on the normal vector of the shading point and use a cosine lobe as distribution which we call the cosine prior:

$$q(\omega_i|n) = \frac{n \cdot \omega_i}{\pi}\,. \tag{4}$$

We motivate the cosine prior because it matches the cosine term in the rendering integral (see Equation 1b and Equation 3), which simplifies the distribution the flow must learn and therefore simplifies the task of the model.

To optimize the model parameters ϕ we minimize the negative log-likelihood between the target density $p(\omega|\mathbf{x})$ and the predicted density $p_\phi(\omega|\mathbf{x})$:

$$L(p, p_\phi) = -\nabla_\phi \mathop{\mathbb{E}}_{\omega\sim p}[\log(p_\phi(\omega))] \tag{5}$$

To avoid sampling the target distribution directly we expand by an-

other density p_s used for sampling training data and expand the negative log-likelihood term:

$$L(p, p_\phi) = -\nabla_\phi \mathop{\mathbb{E}}_{\omega \sim p_s} \left[\frac{p(\omega)}{p_s(\omega)} \log(p_\phi(\omega)) \right] \tag{6}$$

To obtain the predicted density $p_\phi(\omega|\mathbf{x})$ for a given training sample ω we use the fact that each coupling layer is invertible, transforming the sample backward through the composed model, accumulating the absolute Jacobian determinants and finally multiplying by the prior density $q(F_\phi^{-1}(\omega|\mathbf{x}))$ (see Equation 3). To generate training samples we use a simple light tracing algorithm. We used AdamW [LH19] as optimizer with learning rates of 0.005 (Cornell Box) and 0.001 (Veach Ajar & Ganesha).

We implement our model as a proof of concept using the python bindings of Mitsuba 3 [JSR*22], PyTorch [PGM*19] and tiny-cuda-nn [MÜ21]. Unfortunately, the overhead of the python bindings caused high variance, so any timing values gathered were are unrepresentative.

4. Results and Discussion

We evaluate and compare three different prior distributions: a uniform distribution over 3D directions *Uniform Direction*, a uniform distribution in the angular space of spherical coordinates *Uniform Angle*, and our proposed normal-conditioned cosine distribution *Cosine*. The two uniform distributions only differ by the Jacobian term. We differentiate them to investigate whether using a uniform distribution over the 3D directions (the domain of the final output) or over the spherical coordinates (the domain the model works in) changes the results. Since the cosine distribution is isotropic around the normal vector n, sampling it involves only building an arbitrary tangent space around a given normal vector and transforming a local sample to world space.

We evaluate each prior by comparing the *mean average percentage error* (MAPE) over all N pixels of an image, given by Equation 7, where v is the reference and ϵ is a small number to avoid division by zero:

$$\mathrm{MAPE}(v, w) = \frac{1}{N} \sum_{i=1}^{N} \left[\left| \frac{v_i - w_i}{v_i + \epsilon} \right| \right] \tag{7}$$

A comparison of priors in different scenes including MAPE values is shown in Figure 1. In the geometrically simple Cornell Box the difference is small, though the cosine prior shows slightly lower noise in the corners. In the Veach-Ajar scene on the other hand the cosine prior clearly outperforms the uniforms priors, both on the flat table and around the geometrically more complex teapots. In the Ganesha scene we observe similar noise on relatively flat regions but again the cosine prior achieves lower noise in geometrically complex regions.

Figure 2 shows the evolution of the mean average percentage error (MAPE) over the course of training. As already seen in Figure 1, the choice of prior distribution makes little difference in the Cornell Box scene and in fact the simple uniform prior has a slight advantage over the cosine prior. In the Veach-Ajar scene however we observe that the model using the cosine prior initially learns very quickly and generates much less noise. In the Ganesha scene the better initial samples produced by the cosine prior allow for a much better performance from the beginning.

4.1. Faster Convergence

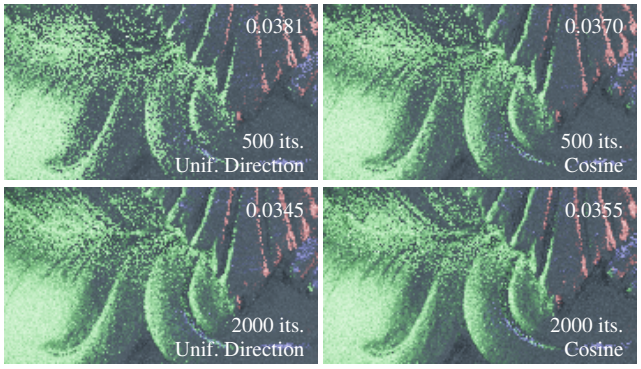

Figure 3: Showing rendered images (32 spp) using models trained for 500 and 2000 iterations. The cosine prior is much better at learning regions with high frequency geometry in 500 iterations compared to the uniform direction prior. We report the MAPE error in the upper-right corner.

In Figure 2 the MAPE evolution in the Ganesha scene shows that the model using the cosine prior starts the training already at a lower error rate and over time the uniform distributions close the gap. This indicates that the uniform priors support learning distributions at locations with complex geometry, but converge slower than the cosine prior which analytically provides geometry information from the beginning.

Rendered images of the Ganesha scene after 500 and 2000 training iterations are presented in Figure 3. We omit the uniform angle prior for clarity since the uniform direction prior performs better on the Ganesha scene at all times during training. These results mirror the MAPE progression in Figure 2: while the cosine prior performs significantly better at 500 iterations, the uniform direction prior eventually closes this gap after longer training times.

4.2. Smaller Models

One motivation for the cosine prior is that it simplifies the task of the model, possibly allowing for smaller models which require less computation time. We compare our small model (section 3) to a larger with 4 total coupling layers, 2 hidden layers for each MLP and $K = 16$ spline segments. Figure 4 shows the convergence plots for the small and large model using the uniform direction and cosine prior respectively on the Ganesha scene. Notably, the difference in sampling quality between the small and large models is significant for the uniform prior while it is small for the cosine prior. Using the cosine prior enables the use of smaller models with minimal impact on accuracy, potentially helping to alleviate the computational overhead of heavy neural path guiding approaches like NIS.

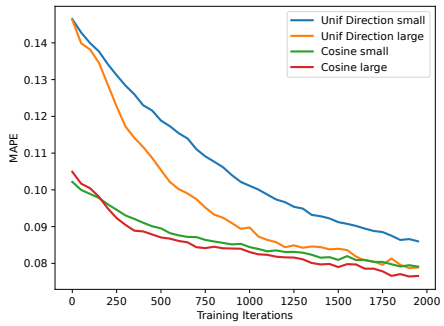

Figure 4: Evolution of MAPE over training with different model sizes on the Ganesha scene. The uniform direction prior performs significantly better when the model is larger and thus more expressive, while the difference is small for the cosine prior.

5. Future Work

One problem both the uniform and cosine priors must overcome is the fact that the splines do not allow samples to wrap around in spherical coordinates since they are strictly monotone. Consider a normal direction pointing closely towards the discontinuity and a light source near it, but entirely on one side. The cosine prior will distribute roughly half of its density to either side of the discontinuity, requiring the model to transport half of the density across the whole domain. In that case the assumption of the cosine prior providing better sample locality is violated, leading to significant direction dependent noise in specific situations.

Since a BSDF can be sampled from and allows arbitrary pdf evaluation the idea of using more expressive prior distributions could be extended to use the current shading points BSDF as prior. Similar to the cosine prior performing well in regions of high geometric detail, a BSDF prior might additionally be able reduce variance in regions of high frequency BSDF details, for example if roughness values are provided by a texture. In addition a BSDF prior would also again support transmissive materials, something the cosine prior does not.

6. Conclusion

In this paper, we have introduced a new cosine-weighted prior distribution for neural path guiding and compared it against standard uniform priors. Our results show that the cosine prior improves convergence, particularly in scenes with high geometric complexity. Furthermore, we demonstrate that choosing a prior that aligns with the rendering integral simplifies the learning task, enabling the use of smaller and faster neural networks without compromising reconstruction quality. This reduction in model complexity helps mitigate the computational overhead of neural path guiding, making it a more practical solution for high-performance rendering and online path guiding.

References

[DBMP19] DURKAN C., BEKASOV A., MURRAY I., PAPAMAKARIOS G.: *Neural spline flows*. Curran Associates Inc., Red Hook, NY, USA, 2019. 2

[DSDB17] DINH L., SOHL-DICKSTEIN J., BENGIO S.: Density estimation using real NVP. In *International Conference on Learning Representations* (2017). URL: https://openreview.net/forum?id=HkpbnH9lx. 1

[DWL23] DONG H., WANG G., LI S.: Neural parametric mixtures for path guiding. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Proceedings* (July 2023), SIGGRAPH '23, ACM, p. 1–10. URL: http://dx.doi.org/10.1145/3588432.3591533, doi:10.1145/3588432.3591533. 1

[FHK25] FIGUEIREDO P., HE Q., KALANTARI N. K.: Neural Path Guiding with Distribution Factorization. In *Eurographics Symposium on Rendering* (2025), Wang B., Wilkie A., (Eds.), The Eurographics Association. doi:10.2312/sr.20251178. 1

[HEV*16] HERHOLZ S., ELEK O., VORBA J., LENSCH H., KŘIVÁNEK J.: Product Importance Sampling for Light Transport Path Guiding . *Computer Graphics Forum* (2016). doi:10.1111/cgf.12950. 1

[HIT*24] HUANG J., IIZUKA A., TANAKA H., KOMURA T., KITAMURA Y.: Online neural path guiding with normalized anisotropic spherical gaussians. *ACM Transactions on Graphics 43*, 3 (Apr. 2024), 1–18. URL: http://dx.doi.org/10.1145/3649310, doi:10.1145/3649310. 1

[JSR*22] JAKOB W., SPEIERER S., ROUSSEL N., NIMIER-DAVID M., VICINI D., ZELTNER T., NICOLET B., CRESPO M., LEROY V., ZHANG Z.: Mitsuba 3 renderer, 2022. https://mitsuba-renderer.org. 3

[Kaj86] KAJIYA J. T.: The rendering equation. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques* (1986), pp. 143–150. 1

[LH19] LOSHCHILOV I., HUTTER F.: Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019* (2019). 3

[MÖ21] MÜLLER T.: tiny-cuda-nn, 4 2021. URL: https://github.com/NVlabs/tiny-cuda-nn. 3

[MGN17] MÜLLER T., GROSS M., NOVÁK J.: Practical path guiding for efficient light-transport simulation. *Computer Graphics Forum (Proceedings of EGSR) 36*, 4 (June 2017), 91–100. doi:10.1111/cgf.13227. 1

[MMR*19] MÜLLER T., MCWILLIAMS B., ROUSSELLE F., GROSS M., NOVÁK J.: Neural importance sampling. *ACM Transactions on Graphics (ToG) 38*, 5 (2019), 1–19. 1, 2

[PGM*19] PASZKE A., GROSS S., MASSA F., LERER A., BRADBURY J., CHANAN G., KILLEEN T., LIN Z., GIMELSHEIN N., ANTIGA L., DESMAISON A., KÖPF A., YANG E., DEVITO Z., RAISON M., TEJANI A., CHILAMKURTHY S., STEINER B., FANG L., BAI J., CHINTALA S.: *PyTorch: an imperative style, high-performance deep learning library*. Curran Associates Inc., Red Hook, NY, USA, 2019. 3

[Vea98] VEACH E.: *Robust Monte Carlo methods for light transport simulation*. Stanford University, 1998. 1

[VKv*14] VORBA J., KARLÍK O., ŠIK M., RITSCHEL T., KŘIVÁNEK J.: On-line learning of parametric mixture models for light transport simulation. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2014) 33*, 4 (aug 2014). 1