

Interactive Hair Rendering and Appearance Editing under Environment Lighting

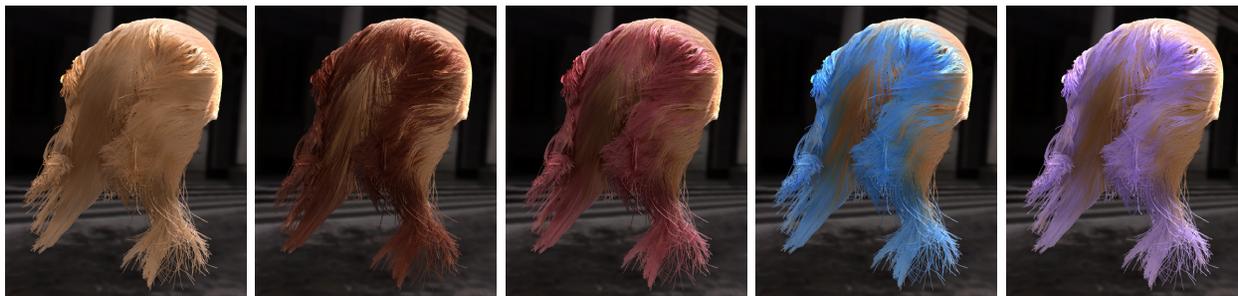


Figure 1: Our algorithm achieves interactive hair rendering and appearance editing under environment lighting, including both single and multiple scattering effects. In this example, the user directly paints onto the hair to edit the spatially-varying scattering parameters. This results in a dynamic simulation of hair coloring. From left to right, the hair is dyed with a progressively more vivid color. The environment map is represented by 40 SRBF lights, and our algorithm runs at 8.3 fps on an NVIDIA GTX 580.

Abstract

We present an interactive algorithm for hair rendering and appearance editing under complex environment lighting represented as spherical radial basis functions (SRBFs). Our main contribution is to derive a compact 1D circular Gaussian representation that can accurately model the hair scattering function introduced by [Marschner et al. 2003]. The primary benefit of this representation is that it enables us to evaluate, at run-time, closed-form integrals of the scattering function with each SRBF light, resulting in efficient computation of both single and multiple scatterings. In contrast to previous work, our algorithm computes the rendering integrals entirely on the fly and does not depend on expensive pre-computation. Thus we allow the user to dynamically change the hair scattering parameters, which can vary spatially. Analyses show that our 1D circular Gaussian representation is both accurate and concise. In addition, our algorithm incorporates the eccentricity of the hair. We implement our algorithm on the GPU, achieving interactive hair rendering and simultaneous appearance editing under complex environment maps for the first time.

Keywords: Hair rendering, environment lighting, appearance editing, SRBFs, circular Gaussian, single and multiple scattering, GPU.

1 Introduction

In hair rendering, it is often desirable to support the dynamic changing of hair’s scattering properties. This enables artists and designers to edit hair appearance at will, and receive realistic rendering feedbacks at interactive speed. Existing methods already support such capability under simple lighting, such as a few point or directional lights [Zinke et al. 2008; Yuksel and Keyser 2008; Shinya et al. 2010; Sadeghi et al. 2010]. However, it remains a challenge to render and simultaneously edit hair appearance under complex lighting such as environment maps [Debevec and Malik 1997]. Such lighting is important to convey the rich look of the hair in natural illumination conditions.

The main difficulty with environment lighting is the large number of directional lights that must be considered. An effective solution is to approximate the environment map as a set of spherical radial basis functions (SRBFs), yielding a low-dimensional representation. This approach has been studied in the context of precomputed light transport and BRDF approximation [Tsai and Shih 2006; Green et al. 2007; Wang et al. 2009]. For hair rendering, Ren et al. [2010]

proposed to integrate the hair scattering function with each SRBF light to produce realistic rendering effects. Their method incorporates both single and multiple scatterings. Unfortunately, as they precompute the integrals of the scattering function with sampled SRBF lights into 4D tables, their method requires fixing hair scattering parameters at precomputation time, disabling online editing.

In this paper, our goal is to enable realistic rendering and simultaneous editing of hair appearance under complex environment lighting. Similar to previous work, we represent an environment map using SRBF lights. Our main contribution is to derive a compact 1D circular Gaussian representation that can accurately model the hair scattering function introduced by [Marschner et al. 2003]. Analyses show that our representation is both concise and accurate. Exploiting the properties of Gaussian functions, the primary benefit of this representation is that it enables the run-time evaluation of closed-form integrals of the scattering function with SRBF lights. This results in efficient computation of both single and multiple scatterings, without the need of expensive precomputation. By using this approach, our algorithm evaluates the rendering integrals entirely on the fly, allowing the user to dynamically change hair scattering parameters at will. Our approach can be seen as an accurate model of the scattering function that is particularly suitable for integration with SRBF lights. In addition, our algorithm successfully incorporates the eccentricity of the hair, which is important for capturing the hair’s rich look.

We provide a GPU implementation that achieves interactive rates for rendering and editing of spatially-varying hair appearance under environment maps. We believe the ability to dynamically adjust the hair appearance under natural lighting provides artists a convenient way for designing hairs. Our method is the first to achieve such capability, including the editing of spatially-varying hair parameters and hair eccentricity effects. Figure 1 shows an example of our results captured at run-time.

2 Related works

Hair Rendering has attracted significant research attention over the years. A comprehensive survey can be found in [Ward et al. 2007]. Some of the early work focused on studying how a single hair fiber reflects the light, which can be described by a hair scattering function. Kajiya and Kay [1989] proposed the first hair scattering function, which has been a simple and popular model in many applications. Marschner et al. [2003] proposed a more accurate model, inspired by the measurements of real human hair fibers. Under dis-

tant lighting, they model each hair fiber as a dielectric cylinder and derive a scattering function that consists of three light paths: R, TT, and TRT (see Fig 2(b)). Zinke and Weber [2007] proposed a more general model that allows for near-field lighting. A number of recent papers have also studied how to estimate hair scattering parameters and geometry from a single photograph [Zinke et al. 2009; Bonneel et al. 2009] or from real hairs using a sophisticated capturing device [Paris et al. 2008; Jakob et al. 2009].

For interactive hair rendering, many existing methods have successfully addressed the issue of self-shadowing in a large number of hair fibers. Some representative techniques include deep shadow maps [Lokovic and Veach 2000], density clustering [Mertens et al. 2004], opacity shadow maps [Kim and Neumann 2001; Sintorn and Assarsson 2008], deep opacity maps [Yuksel and Keyser 2008], and occupancy maps [Sintorn and Assarsson 2009].

As light can bounce many times inside the hair, realistic hair rendering requires the accurate simulation of multiple scattering effects, which are particularly important for light colored hair. Standard path tracing is too expensive for computing multiple scattering, thus researchers have exploited photon mapping [Moon and Marschner 2006; Zinke and Weber 2006] and spherical harmonics approximation [Moon et al. 2008] to greatly improve offline rendering speed. In [2008], Zinke et al. introduced a dual scattering approximation that achieves real-time rendering of multiple scattering. Using this model, Ren et al. [2010] proposed a technique to support interactive hair rendering under environment lighting represented as SRBF lights. This produces rich hair appearance under natural illumination. However, their technique requires fixing scattering parameters offline and ignores the eccentricity of the hair. An alternative way to compute multiple scattering effects is by using a plane-parallel model, as proposed by [Shinya et al. 2010]. While this method is fast under simple lighting, it is still very expensive for rendering under area or environment lights.

Recently, Sadeghi et al. [2010] proposed an artist-friendly control (AFC) model for intuitive editing of hair parameters. Their method assumes simple lighting conditions. As their model is fundamentally based on Marschner’s physically-based scattering function, and the dual scattering approximation, it can be combined with our method in order to provide an intuitive hair editing framework.

SRBF approximates a function defined on the sphere using spherical Gaussians. Compared to other bases such as spherical harmonics and wavelets, SRBF provide excellent local support as well as closed-form solutions for computing product integrals. Therefore it proves to be an excellent choice for representing environment maps and BRDFs [Tsai and Shih 2006; Green et al. 2007; Wang et al. 2009]. Our work uses the circular Gaussian, which is a 1D version of SRBF and thus shares the same benefits. Additionally, it allows us to formulate the integration involving hair scattering functions as 1D integrals, enabling fast and accurate computation.

Material Editing under Environment Maps has drawn a lot of attention in recent years, such as BRDF editing [Ben-Artzi et al. 2006; Sun et al. 2007; Wang et al. 2009] and translucent material editing [Xu et al. 2007; Wang et al. 2008]. As a connection to them, ours is the first to enable hair material editing under environment maps.

3 Background

Terminology. For notations and symbols we follow [Marschner et al. 2003] and [Ren et al. 2010]. Fig 2(a) shows the local geometry of a hair fiber. We use $\omega_i = (\theta_i, \phi_i)$ to denote the lighting direction, and $\omega_o = (\theta_o, \phi_o)$ to denote the viewing direction. All directions

and angles are expressed in the local frame of the hair fiber. Table 1 lists the set of symbols that we use throughout the paper.

Lighting Approximation. As in [Ren et al. 2010] and [Tsai and Shih 2006], an environment map L is approximated as the sum of a set of SRBFs. Using j as the index of an SRBF light:

$$L(\omega_i) \approx \sum_j L_j G(\omega_i; \omega_j, \lambda_j) \quad (1)$$

where L_j is the coefficient, $G(\omega_i; \omega_j, \lambda_j) = \exp(2(\omega_i \cdot \omega_j - 1)/\lambda_j^2)$ is an SRBF centered at ω_j with bandwidth λ_j . In the following we simply write it as $G_j(\omega_i)$. Note that the bandwidth λ_j is defined equivalently but expressed differently from [Ren et al. 2010]. The purpose is to make it consistent with 1D circular Gaussians, which we will use later to approximate the hair’s scattering function.

Single Scattering. Following [Ren et al. 2010], the single scattering radiance viewed from direction ω_o under a set of SRBF lights can be approximated as:

$$L(\omega_o) \approx D \sum_j L_j \tilde{T}(\omega_j, \lambda_j) \int_{\Omega} G_j(\omega_i) S(\omega_i, \omega_o) \cos \theta_i d\omega_i \quad (2)$$

where D is the hair fiber’s diameter, $S(\omega_i, \omega_o)$ is the bidirectional scattering function, and $\tilde{T}(\omega_j, \lambda_j)$ is the effective transmittance, estimated using a summed area table (SAT) built from a convolution optimal depth map. Essentially \tilde{T} measures the average attenuation of an SRBF light j due to transmission of the light through the hair.

Multiple Scattering. Using a dual scattering model [Zinke et al. 2008], Ren et al. [2010] approximate the multiple scattering radiance (denoted as L^D) under environment lighting as:

$$L^D(\omega_o) \approx D \sum_j L_j T_f(\omega_j) \int_{\Omega} \psi_f(\cdot) S^D(\omega_i, \omega_o) \cos \theta_i d\omega_i \quad (3)$$

where $T_f(\omega_j)$ is the forward scattering transmittance along the shadow path of light j , and $S^D(\omega_i, \omega_o)$ is the Bidirectional Curves Scattering Distribution Function (BCSDF). The main idea of dual scattering is to estimate S^D as the sum of a global scattering component and a local scattering component, thus:

$$S^D(\omega_i, \omega_o) = S(\omega_i, \omega_o) + d_b S_{back}(\omega_i, \omega_o) \quad (4)$$

where S is the bidirectional scattering function and S_{back} is the backward scattering function accounting for local scattering. d_b is a constant set between 0.6 – 0.8. Back to Eq 3, $\psi_f(\cdot)$ is the spread function that approximates the final angular distribution of the front scattered radiance, and is computed as:

$$\psi_f(\cdot) = \tilde{s}_f(\phi_j, \phi_i) g^u(\theta_i; \theta_j, \sigma_f(\omega_j, \lambda_j)) / \cos \theta_j \quad (5)$$

where θ_j, ϕ_j are the inclination and azimuth angles of ω_j ; $\tilde{s}_f(\phi_j, \phi_i)$ is defined as $1/\pi$ for forwarding scattering and 0 for backward scattering; and $g^u(\theta_i; \theta_j, \sigma_f)$ is a normalized 1D Gaussian function of θ_i centered at θ_j with bandwidth σ_f . Here $\sigma_f = \sqrt{\lambda_j^2 + \bar{\sigma}_f^2(\omega_j)}$, where $\bar{\sigma}_f^2(\omega_j)$ is the average forward scattering spread.

Summary of [Ren et al. 2010]. As the integrals involved in Eq. 2 and 3 are independent of hair geometry, they can be precomputed into 4D tables by sampling all possible SRBFs offline. At run-time, the effective transmittance \tilde{T} (in single scattering) is approximated by convolution optical depth maps; the forward scattering transmittance T_f and spread σ_f (in multiple scattering) are evaluated by interpolation from sparse samples; and then the two integrals in Eq. 2 and 3 can be estimated by querying the precomputed 4D tables. This allows for interactive hair rendering with 30 ~ 60 SRBF lights. Unfortunately, as the hair scattering coefficients are baked into the precomputed tables, which take hours to compute, dynamic editing of these coefficients is not possible. In addition, it’s unclear how to incorporate the hair eccentricity into their approach.

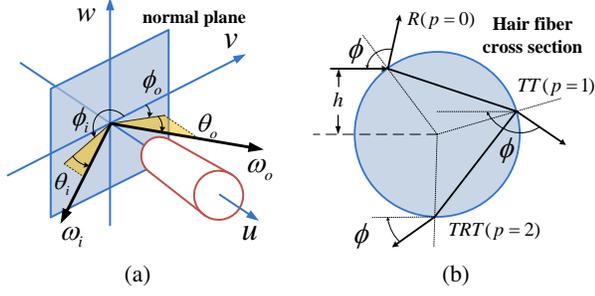


Figure 2: Illustration of (a) notations, and (b) scattering paths.

4 Algorithm Details

Overview. This section described our algorithms. We first review the hair scattering model by Marschner et al. [2003], then derive a compact 1D circular Gaussian representation to approximate the model. We discuss the algorithms for rendering single scattering and multiple scattering separately in Section 4.2 and 4.3.

Key idea. On a high level, our key idea is that an SRBF (i.e. the lighting basis) can be separated into the product of two 1D Gaussians in the two spherical coordinates θ and ϕ (Eq. 11). This matches the formulation of the hair scattering model, which is also written as the product of a longitudinal (θ) function and an azimuthal (ϕ) function (Eq. 6). The separation into 1D functions allows us to quickly compute the rendering integral on the fly, by using the analytic properties of Gaussians and the smoothness of several run-time evaluated terms. Specifically, we exploit the property that the product of two Gaussians is still a Gaussian, and we introduce a simple 1D quadrature method to evaluate the integral of a Gaussian with a smooth function (see Section 4.2.2). Finally, the 1D representation also allows us to precompute certain integrals into small 2D tables, which are used for run-time lookups.

4.1 Hair Scattering Model

Derived in [Marschner et al. 2003], the hair scattering function $S(\omega_i, \omega_o)$ is the sum of three modes: S_R (reflection), S_{TT} (transmission-transmission), and S_{TRT} (transmission-reflection-transmission). Thus $S(\omega_i, \omega_o) = \sum_t S_t(\omega_i, \omega_o)$, where the mode index $t \in \{R, TT, TRT\}$. Each term S_t is further represented as the product of a longitudinal function M_t and an azimuthal function N_t :

$$S_t(\omega_i, \omega_o) = M_t(\theta_h) N_t(\eta, \theta_d, \phi) / \cos^2 \theta_d \quad (6)$$

where θ_h , θ_d and ϕ are the half and difference angles, and η is the index of refraction. The longitudinal function M_t is defined by:

$$M_t(\theta_h) = g^u(\theta_h; \alpha_t, \beta_t) = 2g^u(\theta; 2\alpha_t - \theta_o, 2\beta_t) \quad (7)$$

which is a normalized 1D Gaussian of θ_i centered at $2\alpha_t - \theta_o$ with bandwidth $2\beta_t$. Here α_t and β_t are the longitudinal shift and bandwidth of each mode, caused by the tilting of the hair scales.

The azimuthal function N_t is derived by examining the scattering paths through a circular cross profile, shown in Figure 2(b). First, an analysis of reflections and refractions of a ray through the circle reveals that the azimuthal difference angle $\phi = \phi_o - \phi_i$ can be expressed as a function of the incident direction offset h :

$$\phi(p, h) = 2p \sin^{-1}(h/\eta') - 2 \sin^{-1}(h) + p\pi \quad (8)$$

where η' is the effective index of refraction (see Table 1), p is the number of internal reflections. It is 0 for the R mode, 1 for the TT mode and 2 for the TRT mode. This allows the azimuthal scattering function N_t to be defined as:

η, σ_a	index of refraction, and absorption coefficient
$\alpha_R, \alpha_{TT}, \alpha_{TRT}$	longitudinal shift for R, TT, TRT lobes
$\beta_R, \beta_{TT}, \beta_{TRT}$	longitudinal width for R, TT, TRT lobes
a, w_c	eccentricity, and azimuthal width of TRT lobes
θ, ϕ	longitudinal and azimuthal angles
ω_i, ω_o	incoming and viewing directions
ω_j, λ_j	center and bandwidth of an SRBF light j
$\theta_d = (\theta_o - \theta_i)/2; \theta_h = (\theta_o + \theta_i)/2; \phi = \phi_o - \phi_i; \phi_h = (\phi_o + \phi_i)/2$	
$h \in (-1, 1)$	incident direction offset
$\eta' = \sqrt{[\eta^2 - \sin^2 \theta_d] / \cos \theta_d}$	effective index of refraction
$\lambda_j' = \lambda_j / \sqrt{[\cos \theta_i \cos \theta_j]}$	effective Gaussian bandwidth
$\sigma_a' = \sigma_a / \sqrt{[1 - \sin^2 \theta_d / \eta^2]}$	effective absorption coefficient
$g(x; \mu, \lambda) = \exp[-(x - \mu)^2 / \lambda^2]$	1D Gaussian
$g^u(x; \mu, \lambda) = g(x; \mu, \lambda) / (\sqrt{\pi} \lambda)$	normalized 1D Gaussian
$g^c(x; \mu, \lambda) = \exp[\frac{2(\cos(x - \mu) - 1)}{\lambda^2}]$	1D circular Gaussian
$G(\omega; \omega_j, \lambda) = \exp[\frac{2(\omega \cdot \omega_j - 1)}{\lambda^2}]$	SRBF

Table 1: List of symbols and notations used in this paper.

$$N_t(\eta, \theta_d, \phi) = \sum_h \frac{1}{2} \left\| \frac{d\phi}{dh} \right\|^{-1} A_t(\theta_d, h) \quad (9)$$

which sums over all h 's satisfying $\phi(p, h) = \phi$. Here A_t is given as:

$$\begin{aligned} A_R(\theta_d, h) &= F(\eta, \theta_d, h) \\ A_{TT}(\theta_d, h) &= (1 - F(\eta, \theta_d, h))^2 T(\sigma_a'(\theta_d), h) \\ A_{TRT}(\theta_d, h) &= (1 - F(\eta, \theta_d, h))^2 F(\eta, \theta_d, h) T^2(\sigma_a'(\theta_d), h) \end{aligned} \quad (10)$$

where F is the Fresnel reflection term; σ_a' are the effective absorption coefficient (see Table 1); T is an attenuation factor due to absorption on the internal paths, and is computed by $T(\sigma_a'(\theta_d), h) = \exp(-2\sigma_a' \sqrt{1 - \frac{h^2}{\eta^2}})$.

4.2 Computing Single Scattering

Substitute the scattering mode function defined in Eq 6 to the single scattering integral in Eq 2, and define the result as \mathcal{M}_t , we have:

$$\begin{aligned} \mathcal{M}_t(\omega_j, \lambda_j) &= \int_{\Omega} G_j(\omega_i) S_t(\omega_i, \omega_o) \cos \theta_i d\omega_i \\ &= \int_{\Omega} G_j(\omega_i) M_t(\theta_h) N_t(\theta_d, \phi) \cos \theta_i / \cos^2 \theta_d d\omega_i \end{aligned}$$

By expressing the SRBF G_j in spherical coordinates θ and ϕ , we can decompose it into the product of two 1D circular Gaussians (refer to Appendix A), and expand the integral over Ω to double integrals over θ_i and ϕ_i :

$$\begin{aligned} \mathcal{M}_t(\omega_j, \lambda_j) &= \iint (g_j^c(\theta_i) g_j^c(\phi_i)) M_t(\theta_h) N_t(\theta_d, \phi) \frac{\cos^2 \theta_i}{\cos^2 \theta_d} d\phi_i d\theta_i \\ &= \int g_j^c(\theta_i) M_t(\theta_h) \frac{\cos^2 \theta_i}{\cos^2 \theta_d} \left(\int g_j^c(\phi_i) N_t(\theta_d, \phi) d\phi_i \right) d\theta_i \\ &\approx \int g_j(\theta_i) M_t(\theta_h) \frac{\cos^2 \theta_i}{\cos^2 \theta_d} \mathcal{N}_t(\lambda_j', \phi_o - \phi_j, \theta_d) d\theta_i \end{aligned} \quad (11)$$

where $g_j^c(\theta_i) = g^c(\theta; \theta_j, \lambda_j) = \exp[2(\cos(\theta_i - \theta_j) - 1)/\lambda_j^2]$ and $g_j^c(\phi_i) = g^c(\phi; \phi_j, \lambda_j') = \exp[2(\cos(\phi_i - \phi_j) - 1)/\lambda_j'^2]$ are two 1D circular Gaussians whose product equals $G_j(\omega_i)$. Here the effective azimuthal Gaussian bandwidth $\lambda_j' = \lambda_j / \sqrt{\cos \theta_i \cos \theta_j}$. Note that the last step of Eq 11 uses \mathcal{N}_t to denote the inner integral over

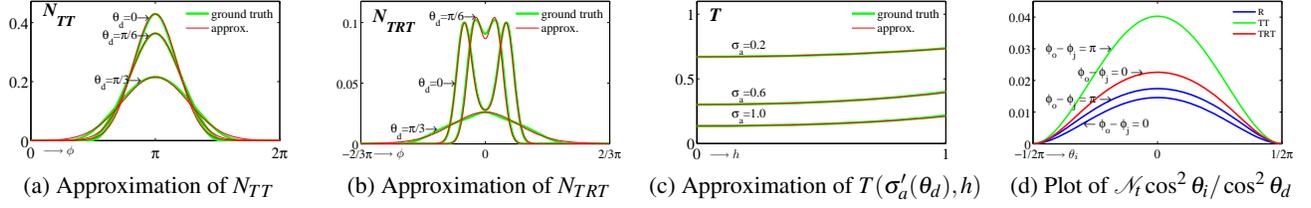


Figure 3: (a): approximating N_{TT} using a single circular Gaussian (Eq 16); (b): approximating N_{TRT} using the sum of three circular Gaussians (Eq 20); (c): approximating $T(\sigma'_a(\theta_d), h)$ using 4-th order Taylor expansion (Eq 18); (d): Plot of $\mathcal{N}_t \cos^2 \theta_i / \cos^2 \theta_d$, $t \in (R, TT, TRT)$.

251 $d\phi_i$; also, it uses the fact that the 1D circular Gaussian $g_j^c(\theta_i)$ is 252 well approximated by a 1D Gaussian $g_j(\theta_i) = g(\theta_i; \theta_j, \lambda_j)$ (proof 253 is shown in Appendix D). By changing the integral variable from ϕ_i 254 to $\phi'_i = \phi_i - \phi_j$, \mathcal{N}_t can be rewritten as:

$$\mathcal{N}_t(\lambda'_j, \phi_o - \phi_j, \theta_d) = \int g^c(\phi'_i; 0, \lambda'_j) N_t(\theta_d, \phi_o - \phi_j - \phi'_i) d\phi'_i \quad (12)$$

255 Essentially Eq 11 has turned the original rendering integral to two 256 1D integrals: the inner integral \mathcal{N}_t and the outer integral \mathcal{M}_t . In the 257 following we will explain how to evaluate these two integrals for the 258 three scattering modes $t \in R, TT, TRT$ respectively.

259 4.2.1 Approximating the R Mode

260 In the R Mode ($p = 0$), the relationship of ϕ and h (Eq 8) is $\phi =$ 261 $-2 \sin^{-1} h$, and hence $h = -\sin(\phi/2)$. Given this, the azimuthal 262 scattering function (Eq 9) can be written as:

$$N_R(\eta, \theta_d, \phi) = \frac{1}{4} |\cos(\phi/2)| F(\eta, \theta_d, -\sin(\phi/2))$$

263 The Fresnel term is defined using Schlick's approximation [1994]: 264 $F(\eta, \theta_d, h) \approx F_0 + (1 - F_0)(1 - \cos \theta_d \sqrt{1 - h^2})^5$, where $F_0 = (1 -$ 265 $\eta)^2 / (1 + \eta)^2$ is the Fresnel value at 0 incident angle. Now N_R can 266 be rewritten as a polynomial of $\cos(\phi/2)$ up to degree 6:

$$N_R(\eta, \theta_d, \phi) \approx \sum_{0 \leq k \leq 6} C_k(\theta_d, \eta) |\cos^k(\phi/2)| \quad (13)$$

267 where each coefficient $C_k(\theta_d, \eta)$ is simply a polynomial of $\cos \theta_d$. 268 Substituting N_R (Eq 13) to the inner integral (Eq 12) yields:

$$\begin{aligned} \mathcal{N}_R &\approx \sum_k C_k(\theta_d, \eta) \int_{-\pi}^{\pi} \left| \cos^k \left(\frac{\phi_o - \phi_j - \phi'_i}{2} \right) \right| g^c(\phi'_i; 0, \lambda'_j) d\phi'_i \\ &= \sum_k C_k(\theta_d, \eta) \mathcal{C}_k(\lambda'_j, \phi_o - \phi_j) \end{aligned} \quad (14)$$

269 where \mathcal{C}_k is the product integral of the k -th power cosine function 270 with a circular Gaussian g^c . Since it only depends on λ'_j and $(\phi_o -$ 271 $\phi_j)$, and not on any scattering coefficient, it can be precomputed as 272 a 2D table and accessed online through table lookups. If the user 273 changes scattering parameters such as η , the coefficients $C_k(\theta_d, \eta)$ 274 will be updated, but this is very fast to evaluate on the fly.

275 Now, let's look back at Eq 11 to see how to calculate the outer 276 integral \mathcal{M}_R . First, note that in that equation, the longitudinal 277 function M_t is a 1D Gaussian given by Eq 7, thus its product 278 with another 1D Gaussian $g_j(\theta_i)$ is still a Gaussian (proof is 279 given in Appendix B). Next, the remaining factor in the integral, 280 $\cos^2 \theta_i / \cos^2 \theta_d \mathcal{N}_R(\lambda'_j, \phi_o - \phi_j, \theta_d)$, is smooth as a function of θ_i . 281 This can be verified from our experiments in Fig 3(d). Thus its 282 product integral with a 1D Gaussian can be accurately estimated 283 using a numerical quadrature as detailed in the section below. In 284 general, the numerical integration only requires a few samplings of 285 \mathcal{N}_R at run-time, which can be quickly evaluated using Eq 14.

286 4.2.2 Product Integral with 1D Gaussian

287 Given a 1D Gaussian $g(x; \mu, \lambda)$ and an arbitrary function $f(x)$, their 288 product integral over the range $[r_0, r_1]$ is defined as:

$$\int_{r_0}^{r_1} g(x; \mu, \lambda) \cdot f(x) dx$$

289 In general, this integral has no analytic solution. However, if $f(x)$ 290 is smooth, it can be numerically estimated using a piecewise linear 291 approximation of $f(x)$. To do so, consider taking $m + 1$ samples x_s , 292 $s \in [0, m]$ and approximate $f(x)$ on each segment $[x_s, x_{s+1}]$ as $f(x) \approx$ 293 $b_s x + c_s$, where b_s and c_s are the linear coefficients calculated from 294 $f(x_s)$ and $f(x_{s+1})$. We can now rewrite the product integral as:

$$\sum_{0 \leq s < m} \int_{x_s}^{x_{s+1}} g(x; \mu, \lambda) \cdot (b_s x + c_s) dx$$

295 which can be easily computed because the product integral of a 296 1D Gaussian with any polynomial function has an analytic solution 297 (this property is shown in Appendix E).

298 Clearly the number of samples $m + 1$ affects the accuracy of 299 this approximation. In our case, since we always integrate over 300 $[-\pi/2, \pi/2]$, we simply compute $m + 1$ samples by picking the 301 first $m - 1$ samples uniformly in the range $(\mu - (m - 2)\lambda/2)$ to 302 $(\mu + (m - 2)\lambda/2)$, plus the two endpoints $-\pi/2$ and $\pi/2$. We have 303 found that choosing $m = 4$ is usually sufficient for the hair scattering 304 function. Fig 4 provides a comparison when using different 305 number of samples. If desired, higher accuracy can be achieved by 306 increasing the number of samples, or using a higher order polynomial 307 approximation.

308 4.2.3 Approximating the TT Mode

309 In the TT mode ($p = 1$), the relationship of ϕ and h in Eq 8 is: 310 $\phi = 2 \sin^{-1}(h/\eta') - 2 \sin^{-1} h + \pi$. From Eq 10 and Eq 9, we have:

$$N_{TT}(\eta, \theta_d, \sigma_a, \phi) = \frac{1}{2} \left| \frac{d\phi}{dh} \right|^{-1} (1 - F(\theta_d, h))^2 T(\sigma_a, \theta_d, h) \quad (15)$$

311 Our experiments in Fig 3(a) show that in practice the function N_{TT} 312 can be very well approximated using a single circular Gaussian 313 centered at $\phi = \pi$, with coefficient b_{tt} and bandwidth λ_{tt} . Thus:

$$N_{TT}(\eta, \theta_d, \sigma_a, \phi) \approx b_{tt}(\eta, \theta_d, \sigma_a) g^c(\phi; \pi, \lambda_{tt}(\eta, \theta_d, \sigma_a)) \quad (16)$$

314 For b_{tt} , we set it directly as the value of N_{TT} at $\phi = \pi$. To set the 315 bandwidth λ_{tt} , note that $\int_{-\pi}^{\pi} g^c(\phi; \pi, \lambda_{tt}) d\phi \approx \sqrt{\pi} \lambda_{tt}$ (see Appendix 316 F), thus we estimate λ_{tt} by preserving the energy of N_{TT} :

$$\lambda_{tt} = \frac{1}{\sqrt{\pi} b_{tt}} \int_{-\pi}^{\pi} N_{TT}(\eta, \theta_d, \sigma_a, \phi) d\phi \quad (17)$$

317 To estimate $\int_{-\pi}^{\pi} N_{TT}$, observe that in Eq 15 the attenuation function 318 T depends on h . Because this function is typically smooth, we can 319 use a 4-th order Taylor expansion to factor out its dependency on h :

$$T(\sigma_a, \theta_d, h) \approx \sum_{k \in \{0,2,4\}} a_k(\theta_d, \sigma_a) h^k \quad (18)$$

where each $a_k(\theta_d, \sigma_a)$ is a Taylor expansion coefficient and has an analytic expression. Fig 3(c) shows the error caused by using the Taylor expansion. From the figure, we can see that the approximation error is generally very small. Combining Eq 18 and 15, we can evaluate $\int_{-\pi}^{\pi} N_{TRT}$ as (note that in the following derivation, we have changed the integral variable from $d\phi$ to dh):

$$\begin{aligned} & \frac{1}{2} \int_{-1}^1 (1 - F(\eta, \theta_d, h))^2 \left(\sum_{k \in \{0,2,4\}} a_k(\theta_d, \sigma_a) h^k \right) dh \\ &= \frac{1}{2} \sum_{k \in \{0,2,4\}} a_k(\theta_d, \sigma_a) \int_{-1}^1 (1 - F(\eta, \theta_d, h))^2 h^k dh \quad (19) \end{aligned}$$

We denote the integral $\int_{-1}^1 (1 - F(\eta, \theta_d, h))^2 h^k dh$ in the above equation as $\mathcal{H}_k^{TRT}(\eta, \theta_d)$. Note that each \mathcal{H}_k^{TRT} ($k = 0, 2, 4$) can be precomputed and stored as a 2D table of η and θ_d .

While we could also estimate λ_{it} by sampling N_{TRT} at a few places, we found our energy-preserving estimation (Eq 17) is more accurate and in fact cheaper to compute on the fly. Now that function N_{TRT} is approximated by a single circular Gaussian (Eq 16), we can substitute N_{TRT} to Eq 12 to evaluate \mathcal{N}_{TRT} . Since the product of two circular Gaussians is still a circular Gaussian, the integral can be easily computed using an analytic solution (see Appendix C and F). Thus we have completed the evaluation of \mathcal{N}_{TRT} .

The outer integral \mathcal{M}_{TRT} (Eq 11) is a product integral between a Gaussian $g_j(\theta_i)$ and function $\mathcal{N}_{TRT} \cos^2 \theta_i / \cos^2 \theta_d$, which is typically smooth. This is in the same form with \mathcal{M}_R discussed in Section 4.2.1, therefore it can be evaluated in the same way as \mathcal{M}_R .

4.2.4 Approximating the TRT Mode

In the TRT mode, the azimuthal function N_{TRT} is more complex. As described by [Marschner et al. 2003], it is symmetric around $\phi = 0$ and contains two peaks when $\eta' < 2$. The two peaks will gradually merge into one single peak centered at $\phi = 0$ when η' increases beyond 2. When $\eta' < 2$, directly using Eq 9 will result in infinite values at the two peaks. Since the hair surface is typically rough, Marschner et al. proposed to replace the two peaks with finite-valued Gaussian functions centered at the peak locations. Fig 3(b) plots N_{TRT} under different θ_d (hence η') values.

Our method employs a similar way to remove the infinite values. As in [Marschner et al. 2003], when $\eta' < 2$, we use two Gaussians centered at the peak locations; however, we use an additional Gaussian centered at $\phi = 0$ to approximate the residue, resulting in a 3-Gaussian approximation. Detailed are discussed below.

When $\eta' < 2$. N_{TRT} is approximated as the sum of 3 Gaussians:

$$N_{TRT} \approx b_1(g^c(\phi; \phi^*, \lambda_1) + g^c(\phi; -\phi^*, \lambda_1)) + b_2 g^c(\phi; 0, \phi^*) \quad (20)$$

where ϕ^* is the peak location calculated as in [Marschner et al. 2003]: $\phi^* = \phi(h^*) = \phi(\sqrt{(4 - \eta'^2)/3})$; $\lambda_1 = w_c$ is the width of peak, which is a user adjustable parameter to control the surface roughness. The coefficient b_2 is set as the residue value at $\phi = 0$:

$$b_2 = N_{TRT}(0)(1 - g^c(0; \phi^*, w_c))^2 \quad (21)$$

We observe that b_2 quickly decreases to zero as η' increases towards 2. For the coefficient b_1 , [Marschner et al. 2003] proposed to estimate b_1 using the local curvature $d^2\phi/dh^2$. Here we propose a different estimation by preserving the energy of N_{TRT} , which is more physically accurate. Specifically, we estimate b_1 as:

$$\frac{\int_{-\pi}^{\pi} N_{TRT}(\phi) d\phi - \int_{-\pi}^{\pi} b_2 g^c(\phi) d\phi}{2 * (\int_{-\pi}^{\pi} g^c(\phi; \phi^*, w_c) d\phi)} \approx \frac{\int_{-\pi}^{\pi} N_{TRT}(\phi) d\phi - b_2 \sqrt{\pi} \phi^*}{2\sqrt{\pi} w_c}$$

where the $\int_{-\pi}^{\pi} N_{TRT}$ term is evaluated at run-time using a similar technique as described in Section 4.2.3. To do so, we first use a 4-th order Taylor expansion of the $T^2(\sigma'_a(\theta_d), h)$ term (Eq 10) to factor out its dependency on h :

$$T^2(\sigma'_a(\theta_d), h) \approx \sum_{k \in \{0,2,4\}} c_k(\theta_d, \sigma_a) h^k \quad (22)$$

where each $c_k(\theta_d, \sigma_a)$ is a Taylor expansion coefficient that has an analytic expression. Next, the integration of N_{TRT} over variable ϕ from $-\pi$ to π can be changed to integrating over variable h from -1 to 1 , which allows us to rewrite $\int_{-\pi}^{\pi} N_{TRT}$ as:

$$\int_{-\pi}^{\pi} N_{TRT}(\phi) d\phi \approx \frac{1}{2} \sum_{k \in \{0,2,4\}} c_k(\theta_d, \sigma_a) \mathcal{H}_k^{TRT}(\eta, \theta_d) \quad (23)$$

where $\mathcal{H}_k^{TRT}(\eta, \theta_d) = \int_{-1}^1 (1 - F(\eta, \theta_d, h))^2 F(\eta, \theta_d, h) h^k dh$ ($k = 0, 2, 4$) is precomputed into a 2D table and accessed on the fly.

When $\eta' > 2$. N_{TRT} is approximated using a single Gaussian: $N_{TRT} \approx b_3 g^c(\phi; 0, \lambda_3)$, where the coefficient b_3 is directly set as the value of N_{TRT} at $\phi = 0$, and λ_3 is determined by preserving the energy similar to the $\eta' < 2$ case. However, this may lead to discontinuity in the representation when η' changes across 2. For example, when $\eta' < 2$, the Gaussian bandwidth is w_c ; and when $\eta' > 2$, the bandwidth is λ_3 . To solve this issue, we follow [Marschner et al. 2003] to use a modified bandwidth that smoothly interpolates between w_c and λ_3 over a small range $\Delta\eta'$, whose value is typically between 0.2 and 0.4. Specifically, the interpolated bandwidth $\lambda'_3 = \text{interp}(w_c, \lambda_3, \text{smoothstep}(2, 2 + \eta', \eta'))$. We also modify b_3 to $b_3 \lambda_3 / \lambda'_3$ to preserve the energy of the lobe.

Summary. When $\eta' < 2$, we approximate N_{TRT} using 3 circular Gaussians; when $\eta' > 2$, we approximate it using one circular Gaussian. The coefficients and centers of the Gaussians are calculated through either analytic functions or query into precomputed 2D tables (\mathcal{H}_k^{TRT}). Once N_{TRT} is represented as circular Gaussian(s), we can proceed to compute \mathcal{N}_{TRT} (Eq 12) as before, using the fact that the product of two circular Gaussians is still a circular Gaussian. Finally, the outer integral \mathcal{M}_{TRT} is evaluated using the piecewise linear quadrature described in Section 4.2.2.

4.2.5 Handling Eccentricity in the TRT Mode

Normally the refractive index η of the hair fiber is set to 1.55. To achieve eccentricity effects due to the elliptical cross sections of real hairs, [Marschner et al. 2003] proposed to use a varying refraction index η^* that depends on the azimuthal half angle ϕ_h :

$$\begin{aligned} \eta_1^* &= 2(\eta - 1)a^2 - \eta + 2, & \eta_2^* &= 2(\eta - 1)a^{-2} - \eta + 2, \\ \eta^*(\phi_h) &= ((\eta_1^* + \eta_2^*) + \cos(2\phi_h)(\eta_1^* - \eta_2^*))/2 \quad (24) \end{aligned}$$

where a is the eccentricity parameter typically ranging from 0.8 – 1.25. To incorporate eccentricity, we update and approximate the integral \mathcal{N}_{TRT} (Eq 12 for the TRT mode) as:

$$\begin{aligned} \mathcal{N}_{TRT} &= \int g^c(\phi'_i; 0, \lambda'_j) N_{TRT}(\eta^*(\phi_h), \theta_d, \phi_o - \phi_j - \phi'_i) d\phi'_i \\ &\approx \int g^c(\phi'_i; 0, \lambda'_j) N_{TRT}(\bar{\eta}^*, \theta_d, \phi_o - \phi_j - \phi'_i) d\phi'_i \quad (25) \end{aligned}$$

where in the last step, we have substituted $\eta^*(\phi_h)$ by its average value $\bar{\eta}^*$, computed as:

$$\overline{\eta^*} = \left(\int g^c(\phi'_i; 0, \lambda'_j) \eta^*(\phi_h) d\phi'_i \right) / \left(\int g^c(\phi'_i; 0, \lambda'_j) d\phi'_i \right) \quad (26)$$

which is efficiently computed on the fly by re-using the precomputed cosine-Gaussian integral tables \mathcal{C}_k prepared for the R mode (see Eq 14). Specifically, using the cosine's double-angle formula, we have $\cos(2\phi_h) = 2\cos^2(\phi_h) - 1 = 2\cos^2(\frac{\phi'_i + \phi_j + \phi_o}{2}) - 1$. Thus we only need to access \mathcal{C}_0 and \mathcal{C}_2 to compute $\overline{\eta^*}$.

The purpose of the $\overline{\eta^*}$ substitution is to remove the dependency of $\eta^*(\phi_h)$ on the integral variable ϕ'_i , which makes it hard to model N_{TRT} as a Gaussian representation. By using the average value $\overline{\eta^*}$, we are able to reproduce convincing eccentricity effects while incurring little modification to our existing algorithm. Fig 5 provides a comparison of our method to the ground truth.

4.3 Computing Multiple Scattering

In this subsection we discuss how to compute multiple scattering. Physically, multiple scattering is more complex than single scattering; however, using the dual scattering approximation [Zinke et al. 2008], multiple scattering is formulated as the sum of a global component and a local component, both of which can be incorporated into our algorithm similarly to single scattering. The key idea is again to exploit the 1D circular Gaussian representations we derived in Section 4.2 for the three scattering modes.

4.3.1 Global Multiple Scattering

According to Eq 3, 4 and 5, the global multiple scattering \mathcal{M}_i^G for each SRBF light j and each scattering mode i is computed as:

$$\begin{aligned} \mathcal{M}_i^G &= \int \psi_f(\cdot) S_i(\omega_i, \omega_o) \cos \theta_i d\omega_i \\ &= \frac{1}{\cos \theta_j} \int \tilde{s}_f(\phi_j, \phi_i) g^u(\theta_i; \theta_j, \sigma_f) M_i(\theta_h) N_i(\theta_d, \phi) \cos \theta_i d\omega_i \\ &= \frac{1}{\cos \theta_j} \int g^u(\theta_i; \theta_j, \sigma_f) M_i(\theta_h) \frac{\cos^2 \theta_i}{\cos^2 \theta_d} \mathcal{N}_i^G(\phi_o - \phi_j, \theta_d) d\theta_i \end{aligned} \quad (27)$$

where the last step expands the integral over $d\omega_i$ to double integrals similar to Section 4.2; and the inner integral is denoted as \mathcal{N}_i^G :

$$\begin{aligned} \mathcal{N}_i^G(\phi_o - \phi_j, \theta_d) &= \int N_i(\theta_d, \phi) \tilde{s}_f(\phi_j, \phi_i) d\phi_i \\ &= \frac{1}{\pi} \int_{\phi_j - \pi/2}^{\phi_j + \pi/2} N_i(\theta_d, \phi_o - \phi_i) d\phi_i \end{aligned} \quad (28)$$

In other words, \mathcal{N}_i^G is just an integral of N_i from $(\phi_j - \pi/2)$ to $(\phi_j + \pi/2)$. This is similar to the single scattering \mathcal{N}_i defined in Eq 12, but is simpler as the g^c term does not appear in the integral.

Thus to compute \mathcal{N}_i^G for each of the R, TT, TRT modes, we can directly use our derivations of N_i from Section 4.2. Specifically: 1) N_R can be expressed as a sum of polynomials of $\cos \phi$ (Eq 13); 2) N_{TT} can be approximated by a single circular Gaussian centered at $\phi = \pi$ (Eq 16); 3) N_{TRT} can be approximated by either one circular Gaussian (when $\eta' > 2$) or the sum of three circular Gaussians (Eq 20, when $\eta' < 2$). Using these approximations, all of N_R , N_{TT} and N_{TRT} have closed form integration formulas, so \mathcal{N}_i^G can be evaluated on the fly.

To evaluate the outer integral \mathcal{M}_i^G , we observe that the product of $g^u(\theta_i; \theta_j, \sigma_f)$ and M_i is still a Gaussian. Therefore we can apply the same method described in Section 4.2.2 to evaluate the quadrature of $\cos^2 \theta_i / \cos^2 \theta_d \mathcal{N}_i^G$. As this function is quite smooth, we found the approximation works very well in practice.

4.3.2 Local Multiple Scattering

The local multiple scattering \mathcal{M}_i^L is computed as:

$$\mathcal{M}_i^L = \frac{1}{\cos \theta_j} \int \tilde{s}_f(\phi_j, \phi_i) g^u(\theta_i; \theta_j, \sigma_f) S_{back}(\omega_i, \omega_o) \cos \theta_i d\omega_i$$

Here the backward scattering function S_{back} is [Zinke et al. 2008]:

$$S_{back}(\omega_i, \omega_o) = \frac{2}{\cos^2 \theta_d} \bar{A}_b(\theta_d) s_b(\phi_i, \phi_o) g^u(\theta_i + \theta_o; \bar{\Delta}_b(\theta_d), \bar{\sigma}_b(\theta_d))$$

where $s_b(\phi_i, \phi_o)$ is $1/\pi$ for backward scattering directions (i.e. $\cos(\phi_i - \phi_o) > 0$), and 0 for forward scattering directions (i.e. $\cos(\phi_i - \phi_o) < 0$). $\bar{A}_b(\theta_d)$, $\bar{\Delta}_b(\theta_d)$, $\bar{\sigma}_b(\theta_d)$ are the average backward attenuation, longitudinal shift and variance respectively. They are all 1D functions of the incident angle θ_d .

Combining the above two equations, we can rewrite \mathcal{M}_i^L as:

$$\begin{aligned} \mathcal{M}_i^L &= \frac{2}{\cos \theta_j} \left(\int \tilde{s}_f(\phi_j, \phi_i) s_b(\phi_i, \phi_o) d\phi_i \right) \cdot \\ &\left(\int g^u(\theta_i; \theta_j, \sigma_f) g^u(\theta_i; \bar{\Delta}_b(\theta_d) - \theta_o, \bar{\sigma}_b(\theta_d)) \bar{A}_b(\theta_d) \frac{\cos^2 \theta_i}{\cos^2 \theta_d} d\theta_i \right) \end{aligned} \quad (29)$$

In other words, it is the product of two 1D integrals. The first integral, which is over $d\phi_i$, can be easily derived analytically:

$$\int \tilde{s}_f(\phi_j, \phi_i) s_b(\phi_i, \phi_o) d\phi_i = (\pi - |\phi_j - \phi_i|) / \pi^2 \quad (30)$$

The second integral, which is over $d\theta_i$, is more complex. We evaluate it as follows. First, the function $g^u(\theta_i; \bar{\Delta}_b(\theta_d) - \theta_o, \bar{\sigma}_b(\theta_d))$ is strictly not a Gaussian, because its parameters (center and bandwidth) depend on θ_d , which in turn depends on the variable θ_i . To remove this dependency, our solution is to substitute $\bar{\Delta}_b(\theta_d)$ and $\bar{\sigma}_b(\theta_d)$ by their mean values computed over $\theta_i \in [\theta_j - \sigma_f, \theta_j + \sigma_f]$:

$$\tilde{\Delta}_b = \frac{\int_{\theta_j - \sigma_f}^{\theta_j + \sigma_f} \bar{\Delta}_b(\frac{\theta_i - \theta_o}{2}) d\theta_i}{2\sigma_f}, \quad \tilde{\sigma}_b = \frac{\int_{\theta_j - \sigma_f}^{\theta_j + \sigma_f} \bar{\sigma}_b(\frac{\theta_i - \theta_o}{2}) d\theta_i}{2\sigma_f} \quad (31)$$

Note that the integration range is essentially the support of the $g^u(\theta_i; \theta_j, \sigma_f)$ function in Eq 29. These integrals can be efficiently computed by using 1D SATs of $\bar{\Delta}_b(\theta_d)$ and $\bar{\sigma}_b(\theta_d)$ built on the fly.

The two mean values above allow us to define a real Gaussian function $g^u(\theta_i; \bar{\Delta}_b(\theta_d) - \theta_o, \tilde{\sigma}_b(\theta_d))$, which will be further multiplied with $g^u(\theta_i; \theta_j, \sigma_f)$ in Eq 29. This results in a new Gaussian, whose product integral with the remaining factor $\bar{A}_b(\theta_d) \cos^2 \theta_i / \cos^2 \theta_d$ can be quickly evaluated using the linear quadrature method described in Section 4.2.2.

Next, we describe how to evaluate the terms $\bar{A}_b(\theta_d)$, $\bar{\Delta}_b(\theta_d)$, and $\bar{\sigma}_b(\theta_d)$ in Eq 29. According to [Zinke et al. 2008], these terms are all defined as functions of $\bar{a}_f(\theta_d)$, $\bar{a}_b(\theta_d)$, $\bar{\alpha}_f(\theta_d)$, $\bar{\alpha}_b(\theta_d)$, $\bar{\beta}_f(\theta_d)$, $\bar{\beta}_b(\theta_d)$, where \bar{a} , $\bar{\alpha}$, $\bar{\beta}$ stand for the average scattering attenuation, shift, and variance; and the subscripts f/b indicate the forward/backward scattering respectively. These definitions can be found in the supplemental material. Using forward scattering as an example: the average forward attenuation \bar{a}_f is computed by integrating the hair's scattering function S over ϕ_d as well as ω_o :

$$\bar{a}_f(\theta_d) = \frac{1}{\pi} \int_{\Omega_f} \int_{-\pi/2}^{\pi/2} S(\theta_d, \phi_d, \omega_o) \cos \theta_d d\phi_d d\omega_o \quad (32)$$

Note that S is the sum of three scattering modes. For each mode t , we expand the above integral by converting the outer integral (over Ω_f) to double integrals over ϕ_o and θ_o :

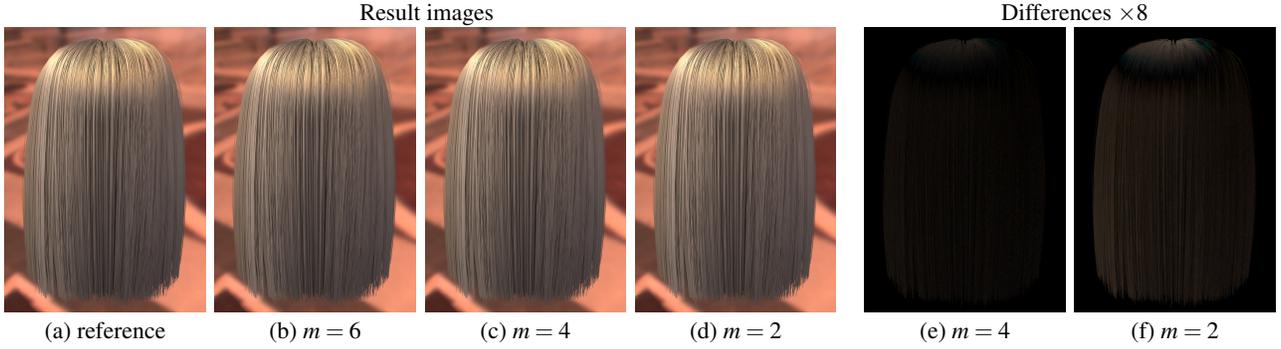


Figure 4: Results of our piecewise linear integration method described in Section 4.2.2. (a) is computed using our method with a large number of integration segments $m=200$, which we treat as a reference; (b,c,d) show the results using a much smaller number of segments: $m = 6, 4, 2$ respectively; (e) and (f) show the magnified difference images of the $m = 4$ and 2 cases with the reference. Note that at $m = 2$, our result has subtle but observable differences compared to the reference; and at $m = 4$, our result is almost indistinguishable from the reference.

$$\bar{a}_{f,t} = \frac{1}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \int_{-\frac{\pi}{2}}^{\frac{3\pi}{2}} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} M_t((\theta_d + \theta_o)/2) N_t((\theta_d - \theta_o)/2, \phi_o - \phi_d) \cdot (\cos \theta_o \cos \theta_d) / \cos^2((\theta_d - \theta_o)/2) d\phi_d d\phi_o d\theta_o$$

488 Noting that scattering occurs at $\theta_o = -\theta_i$, we can substitute $(\theta_d -$
489 $\theta_o)/2$ by θ_d , allowing us to separate the triple integral to:

$$\bar{a}_{f,t} \approx \frac{1}{\pi} \left(\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} M_t((\theta_d + \theta_o)/2) \frac{\cos \theta_o \cos \theta_d}{\cos^2((\theta_d - \theta_o)/2)} d\theta_o \right) \cdot \left(\int_{-\frac{\pi}{2}}^{\frac{3\pi}{2}} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} N_t(\theta_d, \phi_o - \phi_d) d\phi_d d\phi_o \right) \quad (33)$$

490 In the above, the 1D integral involving M_t can be evaluated using
491 our linear quadrature (Section 4.2.2). The 2D integral involving N_t
492 is equal to $2 \int_0^\pi \phi N_t(\theta_d, \phi) d\phi$, the proof of which can be found
493 in the supplemental material. Note that this integral can be ana-
494 lytically computed, again because we have previously modeled N_t
495 either as a sum of polynomials of $\cos \phi$ (R mode) or circular Gaus-
496 sians (TT and TRT modes), whose product integrals with ϕ all have
497 analytic solutions.

498 Finally, the average forward scattering shift $\bar{\alpha}_f$ is calculated as a
499 weighted sum of the longitudinal shifts α_R , α_{TT} , and α_{TRT} using
500 their corresponding attenuation $\bar{a}_{f,t}$ as weights; and the average for-
501 ward scattering variance $\bar{\beta}_f^2$ is the weighted sum of the longitudinal
502 widths β_R , β_{TT} , and β_{TRT} using the same weights. The backward
503 scattering terms (e.g. $\bar{a}_{b,t}$) are similarly computed as in Eq 33, ex-
504 cept for integrating ω_o over the backward scattering directions.

505 Implementation

506 **Preprocessing.** In preprocessing, we follow [Ren et al. 2010] to
507 fit SRBF lights for each environment map. Next, we need to pre-
508 compute several 2D tables, including $\mathcal{C}_k(\lambda'_j, \phi_o - \phi_j)$ ($0 \leq k \leq 6$)
509 for the R mode (Eq 14); $\mathcal{H}_k^{TT}(\eta, \theta_d)$ ($k=0,2,4$) for the TT mode
510 (Eq 19); and $\mathcal{H}_k^{TRT}(\eta, \theta_d)$ ($k=0,2,4$) for the TRT mode (Eq 23).
511 For these tables, we use a resolution of 128 for the parameters λ'_j
512 and η , both of which are sampled logarithmically from the range
513 $\lambda'_j \in [0.001, 10]$ and $\eta \in [1, 10]$ respectively. For $(\phi_o - \phi_j)$ and θ_d
514 we use a resolution of 64. The 3 tables of $\mathcal{H}_k^{TT}(\eta, \theta_d)$ ($k=0,2,4$)
515 are stored as a single 2D texture utilizing the three color chan-
516 nels, so that only one texture lookup is needed to obtain all 3 val-
517 ues. The $\mathcal{H}_k^{TRT}(\eta, \theta_d)$ tables are stored similarly. The 7 tables of

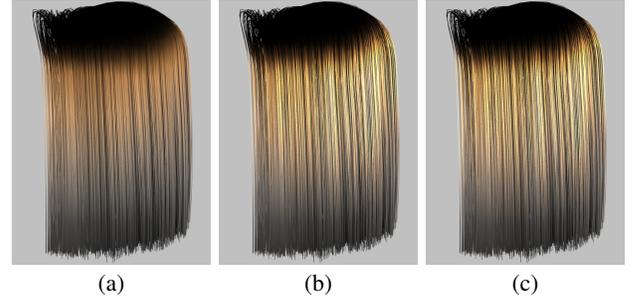


Figure 5: Result of eccentricity approximation. (a) shows circular hair (eccentricity $a = 1$); (b) shows our approximation with eccentricity $a = 0.9$; (c) show a reference with the same eccentricity.

518 $\mathcal{C}_k(\lambda'_j, \phi_o - \phi_j)$ ($0 \leq k \leq 6$) are also stored as a single 2D texture, by
519 placing the values corresponding to $k=1,3,5$ and $k=0,2,4,6$ in adja-
520 cent texels. By utilizing hardware texture interpolation, evaluating
521 \mathcal{N}_R only requires one texture lookup. Note that all these tables are
522 computed only once globally, and do not need to be updated when
523 the user changes hair scattering parameters on the fly.

524 **Runtime Rendering.** Our rendering algorithm is implemented us-
525 ing OpenGL shaders with multi-pass rendering. Each pass renders
526 the result of a subset of 16 SRBF lights, and accumulates contribu-
527 tions from other passes through blending. Hair fibers are rendered
528 as line primitives. For single scattering, the specific steps are:

- 529 1. The effective transmittance $\tilde{T}(\omega_j, \lambda_j)$ (Eq 2) is obtained using
530 an SAT of the convolution optimal depth map [Ren et al. 2010];
- 531 2. For the R mode, the inner integral \mathcal{N}_R is evaluated by texture
532 lookups into $\mathcal{C}_k(\lambda'_j, \phi_o - \phi_j)$ (Eq 14);
- 533 3. For the TT and TRT modes, the fitted circular Gaussian param-
534 eters are first estimated from the two energy terms $\int_{-\pi}^\pi N_{TT}$
535 and $\int_{-\pi}^\pi N_{TRT}$, which are evaluated via texture lookups into
536 $\mathcal{H}_k^{TT}(\eta, \theta_d)$ and $\mathcal{H}_k^{TRT}(\eta, \theta_d)$ (Eq 19 and 23). Following this,
537 each inner integral \mathcal{N}_{TT} and \mathcal{N}_{TRT} is computed as the product
538 of two circular Gaussians using the analytic formula. To account
539 for eccentricity in the TRT mode, a modified index of refraction
540 $\bar{\eta}^*$ is calculated via texture lookups into $\mathcal{C}_k(\lambda'_j, \phi_o - \phi_j)$ (Eq 26),
541 and is used when evaluating \mathcal{N}_{TRT} ;
- 542 4. Finally, we employ the linear quadrature method (Section 4.2.2)
543 to evaluate the outer integrals \mathcal{M}_t (Eq 11). Depending on the
544 number of samples $m + 1$, this requires multiple evaluations of
545 \mathcal{N}_R , \mathcal{N}_{TT} , and \mathcal{N}_{TRT} , which are obtained in steps 2-3.

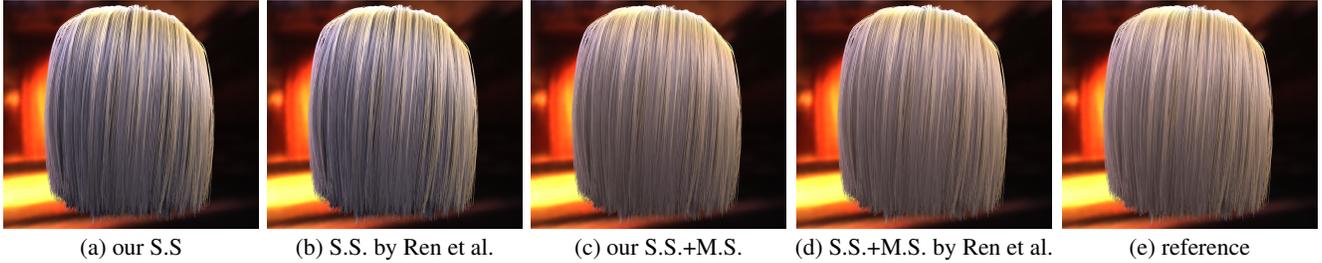


Figure 6: Comparison of our method to [Ren et al. 2010]. (a) and (b) compare the single scattering (S.S.) effect; (c) and (d) compare the full single + multiple scattering (S.S.+M.S.) effect; (e) is a reference image including both single and multiple scattering effects generated using a photon mapping approach [Moon and Marschner 2006]. Note that the images match well visually and the differences are subtle.

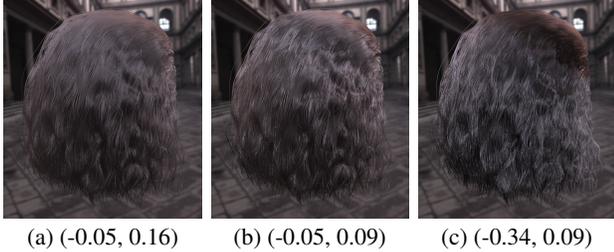


Figure 7: Editing the longitudinal shift and width (α_R, β_R). Note the resulting change in the hair’s highlights.

546 To compute multiple scattering, we first obtain the forward scattering transmittance $T_f(\omega_j)$ and spread $\bar{\sigma}_f^2(\omega_j)$ (Eq 3 and 5) using a sparse sampling approach proposed in [Ren et al. 2010]. We then compute the global and local multiple scattering components separately. Computing the global multiple scattering integral \mathcal{M}_t^G (Eq 27) is very similar to computing the single scattering integral, thus they share largely the same steps as described above.

553 To compute local multiple scattering \mathcal{M}_t^L (Eq 29), we first sample the average terms $\bar{A}_b(\theta_d)$, $\bar{\Delta}_b(\theta_d)$, $\bar{\sigma}_b(\theta_d)$ into a 1D texture of resolution 128, and compute the SATs of $\bar{\Delta}_b(\theta_d)$ and $\bar{\sigma}_b(\theta_d)$. These then allow us to obtain the mean values $\bar{\Delta}_b$ and $\bar{\sigma}_b$ via texture lookups. Finally we use the linear quadrature again to evaluate \mathcal{M}_t^L . Note that the 1D textures are computed and updated on the fly whenever the hair scattering parameters are modified. The total computation overhead is very small.

561 In sum, as we choose $m = 4$ piecewise linear segments, for each SRBF light, computing single scattering at any shading point requires a total of 14 texture lookups, and multiple scattering requires 11 texture lookups. Currently, we do not incorporate the view transparency effect, which can be easily added using techniques such as occupancy maps [Sintorn and Assarsson 2009].

567 6 Comparisons and Results

568 In this section we present results of our method and comparisons to previous work. Many hair geometry models are courtesy of Cem Yuksel et al. [2008], Zinke et al. [2008], and Selle et al. [2008].

571 **Accuracy.** To verify the accuracy of our method, we first mathematically analyze the proposed 1D circular Gaussian representation. In Fig 3 (a,b) we compare our approximated N_{TT} and N_{TRT} functions to the ground truth computed at three different incident angles $\theta_d = 0, \pi/6$ and $\pi/3$. In (c) we compare the 4-th order Taylor expansion of the attenuation factor T to the ground truth computed with three difference absorption coefficients $\sigma_a = 0.2, 0.6$ and 1.0. Note that in all examples, our approximations match the ref-

579 erences very well. We did not plot the N_R function because it does not involve a circular Gaussian approximation.

581 Next, we examine the linear quadrature method described in Section 4.2.2. Recall that we use it to compute all product integrals of 1D Gaussian with a function in the form of $\mathcal{N}_t(\lambda'_j, \phi_o - \phi_j, \theta_d) \cos^2 \theta_i / \cos^2 \theta_d$, where $t \in (R, TT, TRT)$. Examples of these functions, at parameter settings $\lambda'_j = 0.5$ and $\phi_o - \phi_j = 0$ or π , are plotted in Fig 3(d). The cases of N_{TT} at $\phi_o - \phi_j = 0$ and N_{TRT} at $\phi_o - \phi_j = \pi$ are omitted because they are almost zero. As illustrated, these functions are quite smooth, requiring only a few samples (segments) to accurately evaluate their integrals with a Gaussian. The supplemental material provides additional plots of these functions. Fig 4 shows the rendering results computed using our method with different number of segments m , and a comparison to ground truth. Observe that at the default choice of $m = 4$, our result is almost indistinguishable from the reference.

595 In Fig 5 we evaluate the approximated eccentricity described in Section 4.2.4. Our approximation produces visually matching result to the reference, which is computed by using the accurate eccentricity factor and summing up 1536 directional lights in a brute-force way. By accounting for eccentricity, we allow the user to dynamically edit this parameter, resulting in realistic and interesting variations in the hair appearance [Marschner et al. 2003].

602 Finally, in Fig 6, we compare our rendering results to the method presented by [Ren et al. 2010] and the reference generated by photon mapping [Moon and Marschner 2006]. Note that for both single and multiple scattering, the results look qualitatively the same and the differences are subtle. However, our method allows for the dynamic editing of all scattering parameters in the Marschner model, while Ren et al. require fixing these parameters at precomputation time. The supplemental material provides additional comparisons.

610 **Performance.** All results are obtained on a PC with Intel Core 2 Duo 3.00 GHz CPU, 6 GB RAM and an NVIDIA GTX 580 graphics card with 8x anti-aliasing. The image resolution is 720×480 . The rendering frame rates are reported in Table 2. The rendering cost is roughly proportional to the piecewise linear segments m , which we set to 4 by default. In all examples we are able to achieve interactive frame rates.

617 **Parameter editing.** We show several examples of editing hair scattering parameters on the fly. All parameters defined in the Marschner model can be modified. In Fig 1 we show an example where the user paints directly onto the hair model to edit the spatially-varying absorption coefficients. This results in a dynamic simulation of hair coloring computed on the fly. In Fig 7, we edit longitudinal shift α_R and width β_R , and the other parameters $\alpha_{TT} = -\alpha_R/2$, $\alpha_{TRT} = -3\alpha_R/2$, $\beta_{TT} = \beta_R/2$, $\beta_{TRT} = 2\beta_R$ are updated accordingly. This simulates changing the tilting angle the hair scales. In particular, in Fig 7(b), the longitudinal width β_R



(a) (0.03, 0.15) (b) (-0.26, 0.15) (c) (-0.26, 0.3)

Figure 8: Editing the longitudinal shift and width (α_R, β_R).



(a) (b) (c)

Figure 9: editing index of refraction η and the azimuthal width w_c for the TRT caustic. (a) $\eta = 1.54, w_c = 0.06$; (b) $\eta = 1.65, w_c = 0.06$; (c) $\eta = 1.65, w_c = 0.20$.

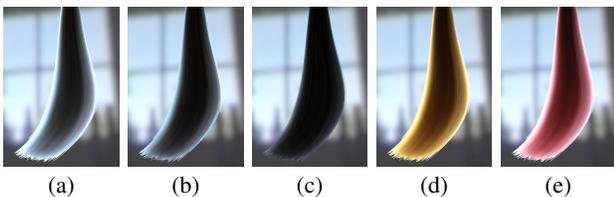
is reduced, resulting in sharper specular highlights. In Fig 7(c), the longitudinal shift α_R is reduced, causing the specular highlights shift downward. Fig 8 gives another example of editing longitudinal shift and width.

Fig 9 shows the editing of the index of refraction and the azimuthal width w_c for the TRT caustic. Observe that 1) increasing η in (b) results in a brighter reflection due to the larger Fresnel reflection term; and 2) increasing w_c in (c) results in smoother specular highlights. Finally, Fig 10 shows the editing of the absorption coefficient, resulting in color changes and back lighting effects caused by the TT component. Please refer to the paper video for additional results.

7 Conclusions

To summarize, we have presented a new method for interactive hair rendering and appearance editing under environment lighting represented as SRBF lights. We derive a compact 1D circular Gaussian representation for the hair scattering function, allowing us to compute their closed-form integrals with SRBF lights at run-time. We provide a GPU implementation that achieves interactive rendering and editing rates for spatially-varying hair parameters and arbitrary environment maps. Such capability is important for design and prototyping applications, and has not been achieved by previous work.

There are several directions for future work. First, we would like to incorporate the artist-friendly model by [Sadeghi et al. 2010]. Since



(a) (b) (c) (d) (e)

Figure 10: Editing the absorption coefficient σ_a . (a) $\sigma_a = (0.36, 0.36, 0.36)$; (b) $\sigma_a = (0.55, 0.55, 0.55)$; (c) $\sigma_a = (1, 1, 1)$; (d) $\sigma_a = (0.27, 0.39, 0.65)$; (e) $\sigma_a = (0.23, 0.42, 0.42)$. Note the change in colors and back lighting effects due to the TT component.

data	#fibers	#points	#SRBFs	FPS
animation (Fig 1)	10k	270k	40	8.3
straight (Fig 6)	50k	1.25M	41	5.9
natural (Fig 7)	10k	1.5M	40	5.8
bob cut (Fig 8)	10k	350k	30	10.6
dark (Fig 9)	15k	1.0M	36	6.2
ponytail (Fig 10)	6k	100k	42	8.9

Table 2: Performance of examples demonstrated in our paper. The timing includes both single and multiple scattering computations.

this model is fundamentally based on Marschner’s model, the dual scattering approximation, and the use cosine/Gaussian functions to approximate the R, TT, TRT azimuthal functions, it can be directly combined with SRBF lighting using our proposed method. Second, we would like to extend our work to near-field light sources in a global illumination environment. Finally, our 1D circular Gaussian representation is not limited to real-time applications – they can also benefit high-quality offline renderings, as they provide a concise and accurate substitute for the original hair scattering function.

References

ABRAMOWITZ, M., AND STEGUN, I. A. 1964. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover, New York.

BEN-ARTZI, A., OVERBECK, R., AND RAMAMOORTHI, R. 2006. Real-time BRDF editing in complex lighting. *ACM Trans. Graph.* 25, 3, 945–954.

BONNEEL, N., PARIS, S., VAN DE PANNE, M., DURAND, F., AND DRETTAKIS, G. 2009. Single photo estimation of hair appearance. *Computer Graphics Forum* 28, 4, 1171C–1180.

DEBEVEC, P. E., AND MALIK, J. 1997. Recovering high dynamic range radiance maps from photographs. In *Proc. of ACM SIGGRAPH*, 369–378.

GREEN, P., KAUTZ, J., AND DURAND, F. 2007. Efficient reflectance and visibility approximations for environment map rendering. *Computer Graphics Forum* 26, 3, 495–502.

JAKOB, W., MOON, J. T., AND MARSCHNER, S. 2009. Capturing hair assemblies fiber by fiber. *ACM Trans. Graph.* 28, 5, 164:1–164:9.

KAJIYA, J. T., AND KAY, T. L. 1989. Rendering fur with three dimensional textures. In *Proc. of ACM SIGGRAPH*, 271–280.

KIM, T.-Y., AND NEUMANN, U. 2001. Opacity shadow maps. In *Proc. of Eurographics Rendering Workshop*, 177–182.

LOKOVIC, T., AND VEACH, E. 2000. Deep shadow maps. In *Proc. of ACM SIGGRAPH*, 385–392.

MARSCHNER, S. R., JENSEN, H. W., CAMMARANO, M., WORLEY, S., AND HANRAHAN, P. 2003. Light scattering from human hair fibers. *ACM Trans. Graph.* 22, 3, 780–791.

MERTENS, T., KAUTZ, J., BEKAERT, P., AND VAN REETH, F. 2004. A self-shadow algorithm for dynamic hair using density clustering. In *SIGGRAPH 2004 Sketches*, 44.

MOON, J. T., AND MARSCHNER, S. R. 2006. Simulating multiple scattering in hair using a photon mapping approach. *ACM Trans. Graph.* 25, 3, 1067–1074.

MOON, J. T., WALTER, B., AND MARSCHNER, S. 2008. Efficient multiple scattering in hair using spherical harmonics. *ACM Trans. Graph.* 27, 3, 31:1–31:7.

PARIS, S., CHANG, W., KOZHUSHNYAN, O. I., JAROSZ, W.,

- 697 MATUSIK, W., ZWICKER, M., AND DURAND, F. 2008. Hair Photobooth: geometric and photometric acquisition of real
698 hairstyles. *ACM Trans. Graph.* 27, 3, 30:1–30:9.
- 700 REN, Z., ZHOU, K., LI, T., HUA, W., AND GUO, B. 2010. In-
701 teractive hair rendering under environment lighting. *ACM Trans.*
702 *Graph.* 29, 4, 55:1–55:8.
- 703 SADEGHI, I., PRITCHETT, H., JENSEN, H. W., AND TAMSTORF,
704 R. 2010. An artist friendly hair shading system. *ACM Trans.*
705 *Graph.* 29, 4, 56:1–56:10.
- 706 SCHLICK, C. 1994. An inexpensive BRDF model for physically-
707 based rendering. *Computer Graphics Forum* 13, 3, 233–246.
- 708 SELLE, A., LENTINE, M., AND FEDKIW, R. 2008. A mass spring
709 model for hair simulation. *ACM Trans. Graph.* 27, 3, 64:1–
710 64:11.
- 711 SHINYA, M., SHIRAISHI, M., DOBASHI, Y., IWASAKI, K., AND
712 NISHITA, T. 2010. A simplified plane-parallel scattering model
713 and its application to hair rendering. *Pacific Conference on Com-*
714 *puter Graphics and Applications*, 85–92.
- 715 SINTORN, E., AND ASSARSSON, U. 2008. Real-time approximate
716 sorting for self shadowing and transparency in hair rendering. In
717 *Proc. of I3D*, 157–162.
- 718 SINTORN, E., AND ASSARSSON, U. 2009. Hair self shadowing
719 and transparency depth ordering using occupancy maps. In *Proc.*
720 *of I3D*, 67–74.
- 721 SUN, X., ZHOU, K., CHEN, Y., LIN, S., SHI, J., AND GUO, B.
722 2007. Interactive relighting with dynamic BRDFs. *ACM Trans.*
723 *Graph.* 26, 3.
- 724 TSAI, Y.-T., AND SHIH, Z.-C. 2006. All-frequency precomputed
725 radiance transfer using spherical radial basis functions and clus-
726 tered tensor approximation. *ACM Trans. Graph.* 25, 3, 967–976.
- 727 WANG, R., CHESLACK-POSTAVA, E., WANG, R., LUEBKE, D.,
728 CHEN, Q., HUA, W., PENG, Q., AND BAO, H. 2008. Real-time
729 editing and relighting of homogeneous translucent materials. *Vis.*
730 *Comput.* 24, 7, 565–575.
- 731 WANG, J., REN, P., GONG, M., SNYDER, J., AND GUO, B.
732 2009. All-frequency rendering of dynamic, spatially-varying re-
733 flectance. *ACM Trans. Graph.* 28, 5, 133:1–133:10.
- 734 WARD, K., BERTAILS, F., KIM, T.-Y., MARSCHNER, S. R.,
735 CANI, M.-P., AND LIN, M. C. 2007. A survey on hair model-
736 ing: Styling, simulation, and rendering. *TVCG* 13, 2, 213–234.
- 737 XU, K., GAO, Y., LI, Y., JU, T., AND HU, S.-M. 2007. Real-time
738 homogenous translucent material editing. *Computer Graphics*
739 *Forum* 26, 3, 545–552.
- 740 YUKSEL, C., AND KEYSER, J. 2008. Deep opacity maps. *Com-*
741 *puter Graphics Forum* 27, 2, 675–680.
- 742 ZINKE, A., AND WEBER, A. 2006. Global illumination for fiber
743 based geometries. In *SIACG 2006*.
- 744 ZINKE, A., AND WEBER, A. 2007. Light scattering from fila-
745 ments. *TVCG* 13, 2, 342–356.
- 746 ZINKE, A., YUKSEL, C., WEBER, A., AND KEYSER, J. 2008.
747 Dual scattering approximation for fast multiple scattering in hair.
748 *ACM Trans. Graph.* 27, 3, 32:1–32:10.
- 749 ZINKE, A., RUMP, M., LAY, T., WEBER, A., ANDRIYENKO, A.,
750 AND KLEIN, R. 2009. A practical approach for photometric
751 acquisition of hair color. *ACM Trans. Graph.* 28, 5, 165:1–165:9.

Appendix

A. Explanation of Eq 11 (Separation of SRBF G_j into two 1D circular Gaussian). First, let's express the directions ω_i and ω_j in Euclidean coordinates:

$$\omega_i = [\sin \theta_i, \cos \theta_i \cos \phi_i, \cos \theta_i \sin \phi_i]$$

$$\omega_j = [\sin \theta_j, \cos \theta_j \cos \phi_j, \cos \theta_j \sin \phi_j]$$

Then, the dot product of ω_i and ω_j can be written as (using product-to-sum rules in trigonometry):

$$\omega_i \cdot \omega_j - 1 = [\cos(\theta_i - \theta_j) - 1] + \cos \theta_i \cos \theta_j [\cos(\phi_i - \phi_j) - 1]$$

Now divide both sides by $\lambda_j^2/2$, and apply the exponential function, and we finally get:

$$G(\omega_i; \omega_j, \lambda_j) = g^c(\theta_i; \theta_j, \lambda_j) \cdot g^c(\phi_i; \phi_j, \lambda_j / \sqrt{\cos(\theta_i) \cos(\theta_j)})$$

B. Product of two 1D Gaussians. A 1D Gaussian g is defined by $g(x; \mu, \lambda) = \exp(-(x - \mu)^2 / \lambda^2)$ where μ is the center and λ is the width. Assume two 1D Gaussians $g_1(x; \mu_1, \lambda_1)$ and $g_2(x; \mu_2, \lambda_2)$, it is straightforward to prove that their product is still a Gaussian, given by $g_1 \cdot g_2 = b \cdot g_3(x; \mu_3, \lambda_3)$, where:

$$\mu_3 = \frac{\mu_1 / \lambda_1^2 + \mu_2 / \lambda_2^2}{1 / \lambda_1^2 + 1 / \lambda_2^2}, \lambda_3 = \sqrt{\frac{1}{1 / \lambda_1^2 + 1 / \lambda_2^2}}, b = e^{-(\mu_1 - \mu_2)^2 / (\lambda_1^2 + \lambda_2^2)}$$

C. Product of two 1D circular Gaussians. A circular Gaussian g^c is defined by $g^c(\phi; \mu, \lambda) = \exp[2(\cos(\phi - \mu) - 1) / \lambda^2]$, which is a function of the angle ϕ on the circle. Assume two circular Gaussians $g_1^c(\phi; \mu_1, \lambda_1)$ and $g_2^c(\phi; \mu_2, \lambda_2)$, it is straightforward to prove that their product is still a circular Gaussian, given by $g_1^c \cdot g_2^c = b \cdot g_3^c(\phi; \mu_3, \lambda_3)$, where:

$$\mu_3 = \arctan(n, m), \lambda_3 = \frac{1}{\sqrt{m^2 + n^2}}, b = \exp\left(-\frac{2}{\lambda_1^2} - \frac{2}{\lambda_2^2} + \frac{2}{\lambda_3^2}\right),$$

$$m = \cos \mu_1 / \lambda_1^2 + \cos \mu_2 / \lambda_2^2, n = \sin \mu_1 / \lambda_1^2 + \sin \mu_2 / \lambda_2^2.$$

D. Approximating circular Gaussian by Gaussian. Using a 2nd-order Taylor expansion: $\cos(\phi - \mu) \approx 1 - (\phi - \mu)^2 / 2$, we can derive that the circular Gaussian $g^c(\phi; \mu, \lambda) \approx \exp(-(\phi - \mu)^2 / \lambda^2) = g(\phi; \mu, \lambda)$. This means a circular Gaussian can be well approximated by a Gaussian. When $\phi \in [\mu - \pi, \mu + \pi]$ and $\lambda < \pi/6$, the relative error of this approximation is $\leq 1.3\%$.

E. Product Integral of 1D Gaussians with Polynomials. First, the indefinite integral of a 1D Gaussian function $g(x; \mu, \lambda)$ is:

$$\int g(x; \mu, \lambda) dx = \frac{1}{2} \sqrt{\pi} \lambda \mathbf{erf}\left(\frac{x - \mu}{\lambda}\right)$$

where \mathbf{erf} is the error function given by: $\mathbf{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$.

It is also known that the indefinite product integral of a Gaussian function with the linear function x is computed as:

$$\int x \cdot g(x; \mu, \lambda) dx = -\frac{1}{2} \lambda^2 g(x; \mu, \lambda) + \frac{1}{2} \sqrt{\pi} \mu \lambda \mathbf{erf}\left(\frac{x - \mu}{\lambda}\right)$$

To evaluate the \mathbf{erf} function, we can either use a precomputed table, or in our case, we use the following approximation [Abramowitz and Stegun 1964] that is easy to evaluate in a shader:

$$\mathbf{erf}(x) \approx 1 - (a_1 t + a_2 t^2 + a_3 t^3) e^{-x^2}, \quad t = 1 / (1 + px)$$

where $p = 0.47047$, $a_1 = 0.3480242$, $a_2 = -0.0958798$, $a_3 = 0.7478556$. This approximation is very accurate, and the approximation error is guaranteed to be smaller than 2.5×10^{-5} .

F. Integral of 1D circular Gaussian. The integral of a 1D circular Gaussian $g^c(\phi; \mu, \lambda)$ over $[-\pi, \pi]$ can be approximated as:

$$\int_{-\pi}^{\pi} g^c(\phi; \mu, \lambda) d\phi \approx \int_{-\pi}^{\pi} g(\phi; \mu, \lambda) d\phi \approx \int_{-\infty}^{\infty} g(\phi; \mu, \lambda) d\phi = \sqrt{\pi} \lambda$$

When $\lambda < \pi/6$, the relative error of this approximation is $\leq 1.9\%$.