# *S(wi)SS*: A flexible and robust sub-surface scattering shader

A. Tsirikoglou[†1], S. Ekeberg[2], J. Vikström[2], J. Kronander[1] and J. Unger[1]

[1]Linköping University, Sweden   [2]Swiss International AB, Sweden



**Figure 1:** *The proposed multi-layered SSS shader enables high quality tunable results. Three separate SSS layers are shown (left), SSS components can be intuitively edited both during rendering and compositing (middle), final composited image (right).*

**Abstract**

*S(wi)SS is a new, flexible artist friendly multi-layered sub-surface scattering shader that simulates accurately sub-surface scattering for a large range of translucent materials. It is a physically motivated multi-layered approach where the sub-surface scattering effect is generated using one to three layers. It enables seamless mixing of the classical dipole, the better dipole and the quantized diffusion reflectance model in the sub-surface scattering layers, and additionally provides the scattering coming of front and back illumination, as well as all the BSDF components, in separate render channels enabling the artist to either use them physically accurately or tweak them independently during compositing to produce the desired result. To demonstrate the usefulness of our approach, we show a set of high quality rendering results from different user scenarios.*

Categories and Subject Descriptors (according to ACM CCS):  I.3.5 [Computer Graphics]:  Computational Geometry and Object Modeling—Physically based modeling

## 1. Introduction

Creating high quality renderings of translucent materials, such as skin, marble, milk, wax etc, is a key aspect in the production of visually interesting images. This, however, presents a significant challenge and requires flexible and highly accurate (physically motivated) subsurface scattering (SSS) modeling. Existing SSS shaders in most current production render engines such as V-Ray, Mental-Ray etc., do not support recently developed highly-accurate SSS models, and provide only a limited support for modeling and tuning materials with multiple subsurface layers (eg. skin consisting of the epidermal, dermal and sub-dermal layers etc.).

In this work, we present *S(wi)SS*, a new artist friendly multi-layered SSS shader that can be used to accurately simulate SSS for a large range of translucent materials. It has been implemented as a shader in V-Ray, and supports the classical dipole [JMLH01, DJ05], the better dipole (BD) [d'E12] and the quantized diffusion reflectance model (QD) [dI11] all in one shader, with artist friendly parameter tuning. Our multi-layered approach enables seamless mixing of the different models in the SSS layers and additionally generates their constituent parts (scattering coming of front and back illumination) in separate render channels enabling the artist to tweak the SSS appearance during both rendering and compositing.

---

† e-mail: apostolia.tsirikoglou@liu.se

## 1.1. Related work

Drawing from the classical diffusion theory [FPW92], and assuming semi-infinite homogeneous media, Jensen et al. [JMLH01] presented the dipole approximation model, a fast approximation for subsurface light transport which combines an accurate single scattering computation with a dipole point source diffusion approximation for multiple scattering. This approximation speeds up the computation of multiple scattering compared to Monte Carlo methods. The dipole model can also be applied to curved surfaces and converts the irradiance at a surface point into two point sources, one above and one below the surface. This approach significantly reduces the complexity of the problem and supports approximate angular incident and exitance variation by factoring the full BSSRDF into a product of Fresnel terms and a radially symmetric function. In this framework, Jensen and Buhler [JB02] presented a hierarchical summation technique to accelerate the computation of the outgoing radiance, since the integration of all incident points contributions is required.

Although the method presented by Jensen et al. [JMLH01] generally produces excellent results for optically-thick and highly scattering materials, the underlying assumptions cause inaccuracies for many geometrically complex objects and exhibit artifacts for semi-transparent or optically-thin materials especially in regions of high-curvature. Donner and Jensen [DJ05] extended the dipole model to a multi-pole one to accurately render thin and multi-layered translucent materials, while a little later [DJ07] they overcome the previous limitations by tracing photons inside the medium to capture inter-scattering between surfaces, and use a quad-pole diffusion approximation to improve accuracy. More specifically, Donner and Jensen [DJ07], placed exponentially attenuated point lights along the refracted light path instead of on the vertical line. This produces elliptical profiles, which are usually observed in reality under oblique lighting directions, but it also requires a numerical integral over the incident light path, which is expensive and prone to sampling noise.

Data-driven techniques have also been used to model BSSRDFs. Such a technique was proposed by Donner et al. [DLR$^*$09] where an empirical BSSRDF is produced for semi-infinite homogeneous materials by fitting Monte Carlo simulated data. These methods use the classical diffusion theory approximation, which suffers from significant errors in near-source and high-absorption cases.

D'Eon et al. [dLE07] proposed an efficient multi-layered model by approximating diffusion profiles using sums of Gaussians. Later, Jimenez et al. [JWSG10] based on this work presented an algorithm for real-time realistic skin-rendering that simulates SSS and extends the screen-space based reflectance approach [JMLH01] by adding transmittance. For this, they used a physically based function that relates the light attenuation with the distance traveled inside the medium. More recently, d'Eon and Irving [dI11] proposed a quantized-diffusion based BSSRDF model. Their method is based on the neutron transport theory and led to several improvements of the diffusion theory. They place an infinite number of point light sources along the vertical incident light path. This requires to compute an integral over the path. The integral is evaluated analytically by approximating the diffusion function with quantized Gaussian functions. They promoted the use of an extended source term instead of approximating it as an impulse at a single depth like the dipole model. Since no closed-form solution exists to the extended source integral, and available numerical approaches were expensive, they approximated the resulting diffusion profile as a sum of Gaussians. To avoid fitting the Gaussians [dLE07], d'Eon and Irving further exploited the fact that time-resolved or quantized diffusion results in a Gaussian distribution and used this as a mathematical basis for finding the Gaussian weights.

Lately, Frisvad, Hachisuka and Kjeldsen [FHK13] presented a new fully analytical model for subsurface scattering that requires no precomputation and uses an approximate solution for a light ray in infinite media. This solution takes into consideration the incident light direction by using two directional sources instead of two point sources, like in classical dipole [JMLH01], relaxing that way the assumption of a flat boundary.

## 2. Background

The layered 2D searchlight problem (Figure 2) is about finding the reflected $R(r)$ and transmitted $T(r)$ energy in a layered scattering medium, as a function of distance $r$ from the point of illumination by a focused beam. The diffuse reflectance profile $R_d(r)$, is used to calculate each shading point color. The implementation of the diffuse reflectance profile is the one that defines the model behavior.

Rendering surfaces requires to compute the light leaving various points on the boundary. For this, Jensen et al. [JMLH01] used Fick's law which states that (for isotropic sources) the vector flux $\vec{E}$ is the gradient of the fluence $\phi$ (flux integrated over time):

$$\vec{E}(r) = -D\vec{\nabla}\phi(r). \qquad (1)$$

In more general terms, this applies Neumann boundary conditions where only derivatives are specified [CC05, HCJ13]. Thus, the diffuse reflectance profile $R_d(r)$ is given by:

$$R_d(r) = C_\phi\phi(r) + C_{\vec{E}}(\vec{E}(r) \cdot \vec{n}), \qquad (2)$$

where $r$ is the distance from the illumination point, $C_\phi$ and $C_{\vec{E}}$ are factors depending on the angular moments and $\vec{n}$ is the normal of the surface where the light arrives.

In classical diffusion theory and due to Fick's law the diffuse reflectance profile $R_d(r)$ depends only on the surface flux (the gradient of the fluence along the surface normal)
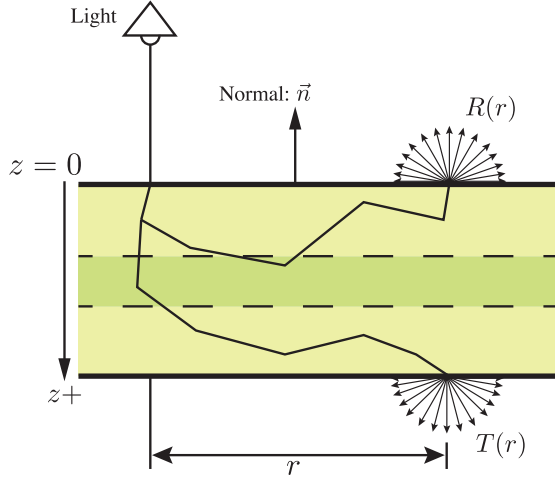
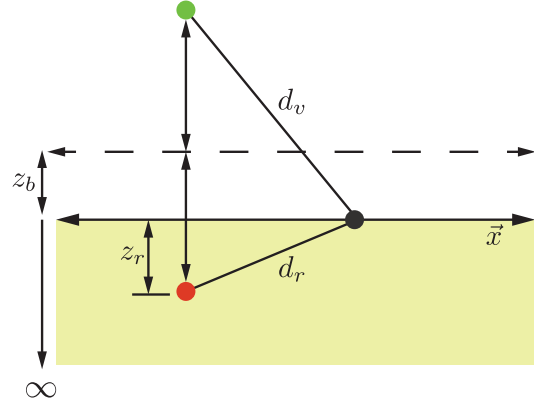**Figure 2:** *The layered 2D searchlight problem.*



**Figure 3:** *The dipole model.*

and not on the fluence $\phi(r)$ itself. Thus, with factors $C_\phi = 0$ and $C_{\vec{E}} = 1$ in eq. (2), so only vector flux and not fluence contributes to the diffuse reflectance, it can be shown that the radiant exitance on the boundary is the dot product of the vector flux with the surface normal:

$$R_d(r) = \vec{E}(r) \cdot \vec{n} = -D(\vec{\nabla} \cdot \vec{n})\phi(r). \qquad (3)$$

Hence, to compute the diffusion profile $R_d(r)$ due to the classical dipole model (Figure 3, where $z_b$ denotes the linear extrapolated distance), the directional derivative $\vec{\nabla} \cdot \vec{n}$ of the fluence in the direction of the normal is needed to be evaluated. This gives:

$$R_d(r) = \frac{\alpha'}{4\pi} \left[ \frac{z_r(1 + \sigma_{tr}d_r)e^{-\sigma_{tr}d_r}}{d_r^3} + \frac{z_v(1 + \sigma_{tr}d_v)e^{-\sigma_{tr}d_v}}{d_v^3} \right] \qquad (4)$$

where $\alpha'$ is the reduced scattering albedo, $z_r$ is the distance of the real light source (placed below the surface) from the surface and $z_r$ the distance of the mirrored virtual light source (placed above the surface) from the surface, $d_r$ is the distance from the ray exit point to the real light source and $d_v$ its distance from the virtual light source, and $\sigma_{tr}$ is the transport coefficient.

The relative contributions of $C_\phi$ and $C_{\vec{E}}$ factors change the exitance calculations in eq. (2) for other boundary conditions.

D'Eon [d'E12] introduces the better dipole assuming that the multiple scattering term can be given by a diffusion method of images solution to the 2D searchlight problem. He furthermore simplifies the searchlight problem assuming that the light arrives normal to the surface and computing

only the net flux leaving the surface at a distance $r$ from the illumination point. The diffuse reflectance profile $R_d(r)$ for the better dipole model is then given by

$$R_d(r) = \frac{(\alpha')^2}{4\pi} \left[ \left( C_{\vec{E}} \frac{z_r(1 + \sigma_{tr}d_r)}{d_r^2} + \frac{C_\phi}{D} \right) \frac{e^{-\sigma_{tr}d_r}}{d_r} - \left( C_{\vec{E}} \frac{z_v(1 + \sigma_{tr}d_v)}{d_v^2} + \frac{C_\phi}{D} \right) \frac{e^{-\sigma_{tr}d_v}}{d_v} \right] \qquad (5)$$

where $D$ is the diffusion coefficient and the fluence and flux factors are given now by

$$C_\phi = \frac{1}{4}(1 - 2C_1) \qquad \text{and} \qquad C_{\vec{E}} = \frac{1}{2}(1 - 3C_2)$$

respectively instead of classical dipole's $C_\phi = 0$ and $C_{\vec{E}} = 1$, and $C_1$ and $C_2$ are angular moments which approximations are given by D'Eon and Irving [dI11].

Finally, D'Eon and Irving [dI11] proposed the quantized diffusion model, where the diffuse reflectance profile is approximated by a sum of Gaussians:

$$R_d(r) \approx \alpha' \sum_{i=0}^{k-1} w_R(i)w_i G_{2D}(v_i, r) \qquad (6)$$

where $w_R(i)$, $w_i$ are weights and $v_i$ are the variances of normalized 2D Gaussians $G_{2D}$. The equations necessary to calculate the weights and variances of the Gaussians are themselves summations of integrals depending on further weights $w_{\phi R}(v, i)$ and $w_{\vec{E}R}(v, i)$.

## 3. Multi-layer sub-surface scattering

*S(wi)SS* creates a layered BSDF based on three components: (1) a bottom SSS component, (2) a middle diffuse component and (3) a top specular component.
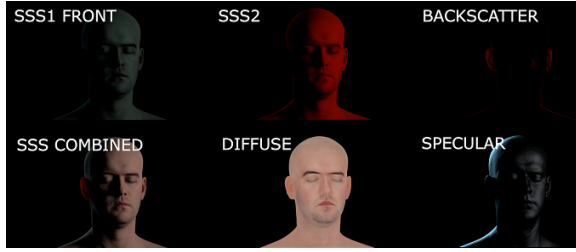
**Figure 4:** *BSDF's SSS, diffuse and specular components of our multi-layer approach 3.*

In Figure 4 the BSDF components are shown. The bottom SSS component (SSS COMBINED) is produced by the combination of three SSS layers (SSS1 FRONT, SSS2 and BACKSCATTER). The SSS component is then combined with the diffuse component and finally with the specular one.

(1) The bottom SSS component consists of a basic SSS layer providing front, (optionally) back, and single scattering. A second SSS layer provides front scattering, and a final SSS layer for back scattering.
Each of the three SSS layers has its own scatter color and radius and can be weighted separately. In each SSS layer, the user is able to apply the classical dipole, eq. (4), the BD, eq. (5), or the QD, eq. (6), model. Thus, the rendering time can be adjusted by applying, for example, the most-detailed and time consuming QD to the top SSS layer and the faster but less-detailed classical dipole or BD to the remaining SSS layers.
The second and third SSS layers do not significantly contribute to single scattering and thus only support multiple scattering to reduce rendering time.

(2) The middle diffuse component is a classic Lambertian reflection model. Similar to the SSS component, the diffuse component contribution can also be manually adjusted through the user interface. The SSS and diffuse components are combined before layered in the BSDF.

(3) The surface specular component consists of two Phong models for specular reflection. Two specular layers are used to simulate looks such as the upper, oily layer of skin atop the regular specular layer. Each of them provides separate specular color, weight, glossiness, subdivisions and index of refraction settings. Like the rest of the BSDF components, the specular one's contribution can also be weighted through the shader interface.

### 3.1. Front and back illumination map

We use an illumination map to calculate the color of each shading point. Front and back scattering are separated by classifying the illumination map samples in two categories: the ones on the front side of the object and the ones on the back side with respect to the camera view. During rendering,
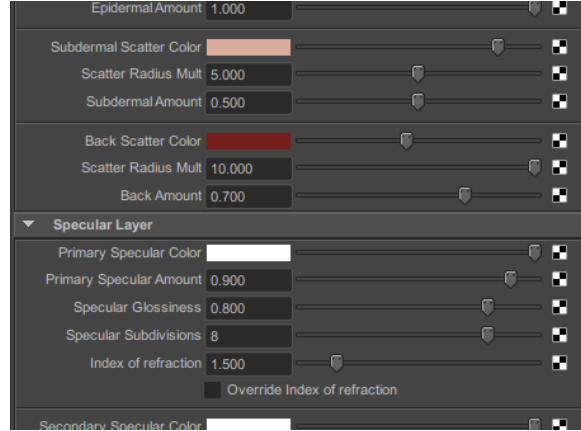


**Figure 5:** *SSS and specular settings in a representative shader view 4.*

the front and back illumination is weighted according to the applied model (BD or QD).

This way, the user can totally control the SSS component, and its separate front and back SSS layers and therefore to handle them physically accurately or tweak them independently according to the specific needs.

### 4. Implementation

We implemented our algorithm in V-Ray 2.0 by Chaos Group for Autodesk's Maya but a similar implementation would be possible to use with other render engines as well.

We based the generation of the illumination map for *S(wi)SS* on the existing mechanism in *VRayFastSSS2*, the SSS shader packaged with V-Ray. The front and back scattering layers are generated in separate render channels enabling the artist to tweak them separately, or rely on physical accuracy to determine the final look.

During rendering, the diffuse reflectance profile, Rd, is used to calculate each shading point's color. For the BD model, we improved VRayFastSS2's existing diffuse reflectance profile by implementing the method described by d'Eon [d'E12].

QD is based on the improved diffusion theory with Grojean's approximations and uses a sum of weighted 2D Gaussians to approximate the reflectance profile [dI11]. An extended multipole model for a finite material of finite thickness was set up instead of a dipole. The quantization parameters regarding time, variances and weights are initialized, calculated and saved for later use during rendering when the diffuse reflectance profile is evaluated by the sum of Gaussians. These pre-calculated parameters significantly reduce rendering time.

*S(wi)SS* provides several settings to modify properties like

the material index of refraction and light map options (eg. the illumination map can be built by an irradiance map or geometry sampling). Moreover, the user can intuitively control all the BSDF components and their layers 3 through the shader interface by setting the desired amounts, colors and the specific options given for each component. In Figure 5 it is given a representative view of the user interface and how the layers of the SSS and specular components can be set. The render elements that can be moreover tuned during compositing phase consist of all three BSDF components (SSS, diffuse and specular), as well as their layers including the scattering coming of front and back illumination. This way, the artist is enabled to tweak (enhance or weaken) any component according to the specific look demands. For example, in skin rendering it is usually desired an enhanced back scattering layer for strong SSS effects and thus highly realistic results.

## 5. Results

This section presents a set of results rendered using the *S(wi)SS* shader. The rendered images compare and highlight the effect of the artist friendly shader parameters on the final results. We also show figures describing the performance of our implementation. The renderings are mainly focused on the models comparison and different algorithm combinations.

In Figure 6 the final result using the BSDF components shown in Figure 4 is given. The model is illuminated from both the front and the rear. The SSS component is created using all three SSS layers, where the QD model, eq. (6), is applied to the basic SSS layer, and the BD model, eq. (5) is applied to the rest SSS layers. For better SSS algorithm evaluation, we show in Figure 12 the SSS component as it is given by the classical dipole, the BD and QD models. The level of the rendered details is increased from left (classical dipole) to right (BD and rightmost QD). Please note areas like the lips, eyes and ears, where the differences and the SSS effect are clear.

In Figure 8 a 3D model, similar to the one that d'Eon and Irving [dI11] used to present their QD method, has been rendered just with the first basic SSS layer of the SSS component using the classical dipole (*VRayFastSSS2*), eq. (4), BD, eq. (5), and QD, eq. (6), algorithms (from left to right) for easy side-by-side comparisons. In Figure 9 detailed views of the model comparative results in Figure 8 are given. Their rendering times are given in Table 1. All the images were rendered with an Intel Xeon W3680, 12 MB Cache, 3.33 Ghz, Six-Core server processor machine. BD is almost as fast as *VRayFastSSS2*, that implements the classical dipole method, whereas the QD model is slower. This is expected as the implementation of QD requires the sum of 36 approximately weighted Gaussians for every shading point. Moreover, the above given times are for hight quality renderings but it can be reduced by decreasing the resolution of surface



**Figure 6:** *Final result using the BSDF components shown in Figure 4.*

lighting, computed during the prepass phase. Regarding the visual result, the dominance of QD is obvious. The level of the rendered details is significantly higher than the classical and BD providing at the same time realistic subsurface scattering effects. Moreover, QD gives the smoothest shading result in the rendered surface.

| Method | Time per frame | |
|---|---|---|
| Classical Dipole | $\sim$ | 43sec |
| Better Dipole | $\sim$1min | 2sec |
| Quantized Diffusion | $\sim$6min | 2sec |

**Table 1:** *Average rendering times for renderings in Figure 9.*

In Figure 10 the front and back separation can be seen. In the particular rendering, only the first basic SSS layer is used, where the QD model, eq. (6), is applied (rightmost rendering in Figure 8). The SSS layer is then given by the combination of the produced front and back scattering.

In Figure 11 multi-layer SSS rendering results are shown. To the left, the QD model, eq. (6), was used just for the basic SSS layer. In the middle the QD model was used for the basic SSS layer and the BD, eq. (5) for the second SSS layer, whereas to the right the QD model was used for the basic SSS layer, the BD for the second SSS layer and the classical dipole, eq. (4), for the final back SSS layer. It is worth noticing that as we add more layers to the SSS component the result changes depending on the SSS layer's scatter color and radius. Specifically in Figure 11 the added SSS layers scatter mainly the red color, explaining the respective rendered results.

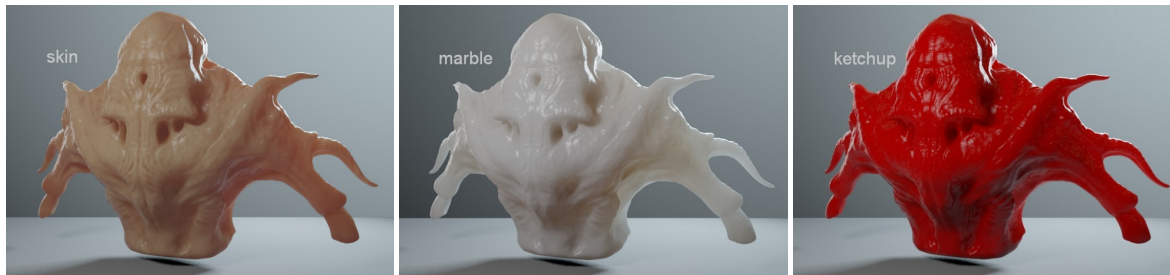Finally, in Figure 7 renderings for different translucent

**Figure 7:** *Skin (left), marble (middle) and ketchup (right) rendered with S(wi)SS shader.*

materials are presented. For skin rendering (left) we used all three layers of the SSS component with the QD model, eq. (6), for the basic SSS layer and the BD one, eq. (5), for the rest two, whereas for the marble (middle) and ketchup (right) renderings only the basic SSS layer was used where the QD model, eq. (6) was applied.

## 6. Discussion

*S(wi)SS* is a powerful and flexible tool that compared to the existing commercial shaders uses recently introduced highly-accurate SSS calculations. It is related with the corresponding implementations coming packaged with V-Ray and Mental ray. However, our multilayer approach provides, additionally to the standard options, advanced approximations for detailed SSS for each layer of the SSS component and generates all the BSDF components including scattering coming of front and back illumination in separate render channels all in one rendering.

## 7. Conclusion

*S(wi)SS* creates highly detailed SSS using one to three layers. The artist can use the separate render elements either physically accurately or tweak them independently during the compositing phase. Figure 1 represents a typical *S(wi)SS* workflow where the artist adjusts the SSS render elements in compositing software, synthesizing the desired look.

Responding to the demands in visual effects production, *S(wi)SS* is a flexible, accurate, easy-to-use look-development tool that introduces a multi-layered SSS approximation to simulate different material types and provides enhanced user control. The S(wi)SS shader and its surrounding process is currently being used effectively in a visual effects studio.

## References

[CC05]  CHENG A. H.-D., CHENG D. T.:  Heritage and early history of the boundary element method. *Engineering Analysis with Boundary Elements 29*, 3 (2005), 268–302. 2

[d'E12]  D'EON E.:  A better dipole.  `http://www.eugenedeon.com/papers/betterdipole.pdf` (2012). 1, 3, 4

[dI11]  D'EON E., IRVING G.: A quantized-diffusion model for rendering translucent materials.  *ACM TOG 30*, 4 (July 2011), 56:1–56:14. 1, 2, 3, 4, 5

[DJ05]  DONNER C., JENSEN H. W.:  Light diffusion in multi-layered translucent materials.  *ACM TOG 24*, 3 (July 2005), 1032–1039. 1, 2

[DJ07]  DONNER C., JENSEN H. W.:  Rendering translucent materials using photon diffusion.  In *Rendering Techniques* (2007), pp. 243–251. 2

[dLE07]  D'EON E., LUEBKE D., ENDERTON E.:  Efficient rendering of human skin.  In *Proc. of Eurographics Symposium on Rendering* (2007), pp. 147–157. 2

[DLR*09]  DONNER C., LAWRENCE J., RAMAMOORTHI R., HACHISUKA T., JENSEN H. W., NAYAR S.:  An empirical bssrdf model. *ACM TOG 28*, 3 (July 2009), 30:1–30:10. 2

[FHK13]  FRISVAD J. R., HACHISUKA T., KJELDSEN T. K.:  Directional dipole for subsurface scattering in translucent materials. *ACM TOG* (Aug. 2013). Manuscript. 2

[FPW92]  FARRELL T. J., PATTERSON M. S., WILSON B.: A diffusion theory model of spatially resolved, steady-state diffuse reflectance for the noninvasive determination of tissue optical properties in vivo. *Medical Physics 19*, 4 (1992), 879–888. 2

[HCJ13]  HABEL R., CHRISTENSEN P. H., JAROSZ W.:  Classic and modified diffusion theory for subsurface scattering. *Tech. rep.* (2013). Disney Research Zürich. 2

[JB02]  JENSEN H. W., BUHLER J.:  A rapid hierarchical rendering technique for translucent materials.  In *Proc. SIGGRAPH 2002* (2002), pp. 576–581. 2

[JMLH01]  JENSEN H. W., MARSCHNER S. R., LEVOY M., HANRAHAN P.: A practical model for subsurface light transport. In *Proc. SIGGRAPH 2001* (2001), pp. 511–518. 1, 2

[JWSG10]  JIMENEZ J., WHELAN D., SUNDSTEDT V., GUTIERREZ D.: Real time realistic skin translucency. *IEEE CG&A 30*, 4 (2010), 32–41. 2

**Figure 8:** *S(wi)SS shader renderings using just the first basic SSS layer of the SSS component implementing the classical dipole (left), BD (middle) and QD(right).*
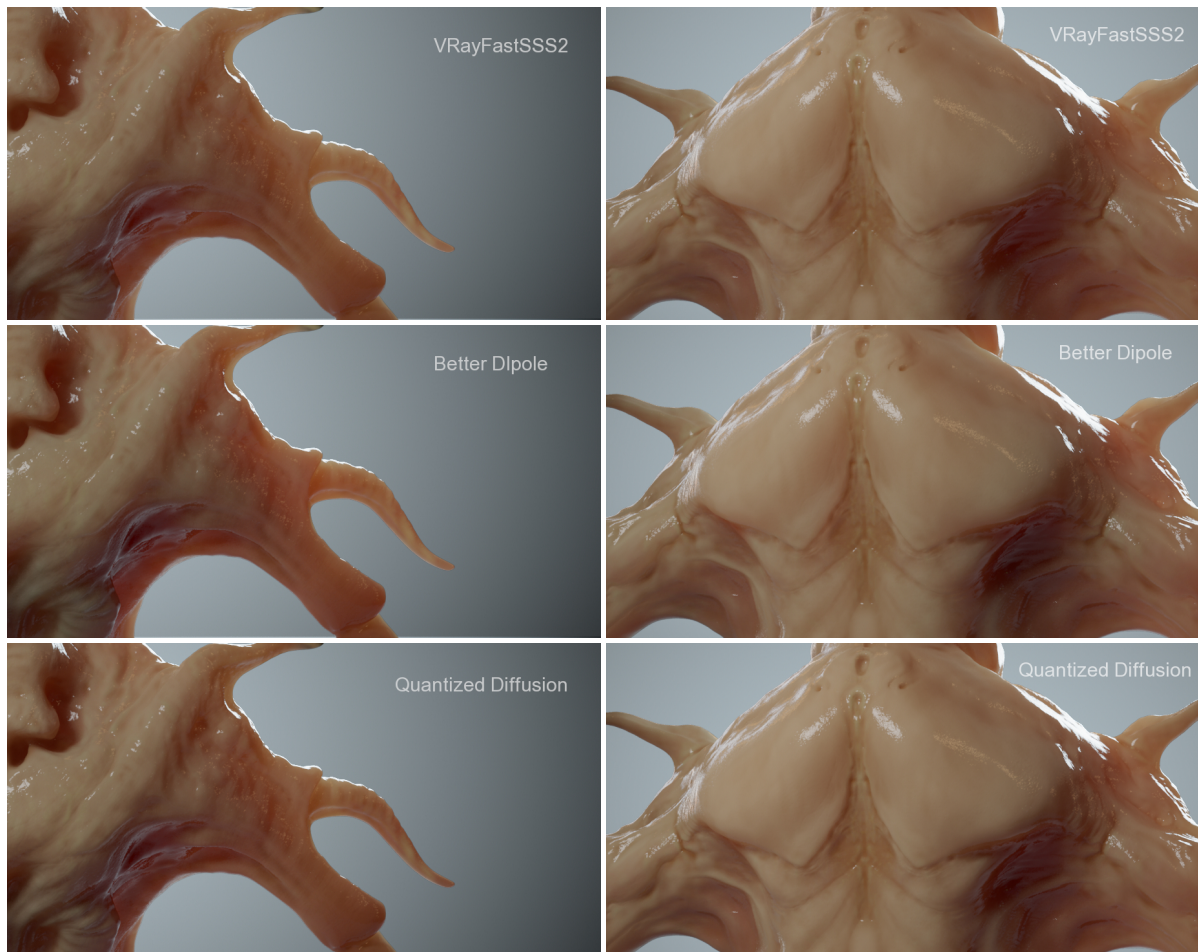

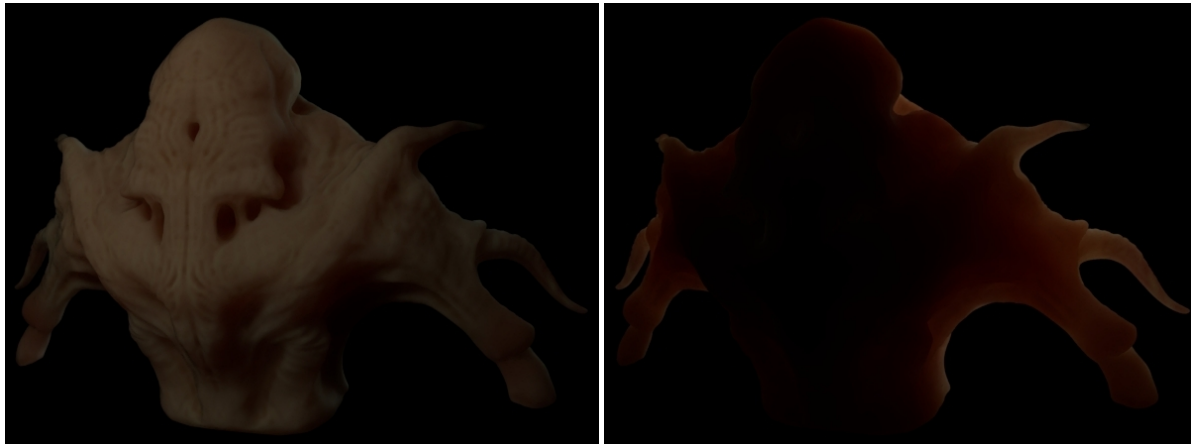
**Figure 9:** *Detailed views of renderings in Figure 8.*

**Figure 10:** *Front and Back scattering separation in QD model.*



**Figure 11:** *S(wi)SS renderings with one (left), two (middle) and three (left) SSS layers, using the QD model as the basic SSS layer, the BD as the middle SSS layer and the classical dipole as the last back SSS layer respectively.*



**Figure 12:** *SSS component rendered with the classical dipole (left), the BD (middle) and QD (right) model.*