

# Skin Rendering by Pseudo–Separable Cross Bilateral Filtering

Morten S. Mikkelsen  
Naughty Dog Inc., USA

August 28, 2010

## Abstract

This paper proposes a real–time solution to achieve results which are visually reminiscent of human skin and may be applied to fully deformable objects.

A similar method has been proposed which applies a regular Gaussian blur in screen–space. It is reported that this results in errors near the silhouette of the surface. In our work we provide a more vigorous derivation which shows that surface convolution by a Gaussian function can be expressed as a cross bilateral filter in screen–space which solves the reported errors.

The method operates as a post–process on an irradiance buffer recorded in screen–space and is thus light model independent and executes only once per frame.

## 1 Introduction

The visual characteristics of a material are due to the interaction between light and the molecular structure and density of the medium. As an example metal distinguishes from a soft material because it does not exhibit any translucency due to *subsurface scattering*. This phenomenon occurs when light enters the medium and scatters around internally. The light will either be absorbed inside the medium or travel back out to the boundary of the medium and leave it while moving in some final outgoing direction. This process blurs the reflected light by wavelength–dependent profiles.

To capture the appearance of translucent materials, light transferred beneath the surface between surface points must be taken into account. This is true for many materials such as fruits, marble, ice, plants, and skin. Technically, this requires numerically solving the more general equation of transfer which is volumetric.

## 2 Previous Work

In 2001 Henrik Wann Jensen et al. [JMLH01] introduced a technique for rendering skin using an approximate reflectance profile designed by Patterson et. al [PCW89] of the medical physics community. The method allows us to determine the outgoing radiance by integrating across the surface as opposed to a costly volumetric evaluation. Furthermore, the reflectance profile is equipped with an exponential fall off. Both properties enable us to accelerate the integration process.

A method to perform this integration in software was suggested by Jensen in 2002 [JB02] where sample points are evenly distributed across the surface of an object. The rapid fall off of the reflectance profile is exploited by using an octree structure to allow for a more crude precision during integration at remote sample points.

The solution works well but does not offer interactive performance. Since then, several proposed methods have been introduced to avoid per pixel octree traversal by adapting the integration process to an image-space. One example was introduced in [BL03] to render faces. As opposed to distributing samples across the actual surface of the mesh one might approximate by performing the integration based on distances between texels and thus complete the integration in a two dimensional unwrap. Disadvantages are that integration must be done per visible character and the method does not account for transmittance through thin regions such as ears and nostrils.

The work of [BL03] was adapted to a more GPU friendly implementation by Eugene d'Eon et al. in [dL07]. This was done by assuming the reflectance profile can be expressed by a sum of Gaussians and thus take advantage of separability. An addition, to the solution was proposed such that transmittance is accounted for. However, this works by assuming that each source is either a point or a directional light. Furthermore, integration has to be done per source.

Others have used screen-space implementations. Examples are the work of Kasper H. Nielsen, at IO-Interactive, and also the independently developed and published work of [JSG09] where a 2D Gaussian filter is used. The method does take advantage of separability and adjusts the filter size based on slope and distance. However, it is reported that there are errors which appear as a halo near the silhouette. This problem is solved by the method which is derived in the following.

## 3 Integration by Substitution

A problem with a screen-space solution is that we only have samples for front-facing primitives. This is sufficient to account for *local subsurface scattering* but does not support light transmitted through thin materials.

An approach, which solves this problem, is given by an unpublished subsurface scattering implementation [vL07] for Blender. Their solution renders out two irradiance buffers in screen-space (seen from the camera). Front-facing

primitives in the first buffer and back-facing in the second. Subsequently, the surface point, corresponding to each pixel, is inserted into an octree and the process continues in software as in [JB02]. Thus the initial distribution of sample points is achieved simply by rendering to the screen. This gives a natural correspondence in sampling density between the irradiance image and the final output. A lack of sampling density will occur near the silhouette edge as seen from the camera, but details are faded here because the ratio between the projected area and the actual surface area is small.

We seek a solution which is light model independent and can be performed efficiently on the GPU. Adaptively building an octree on the GPU is relatively inefficient. A possible remedy is to build a custom mip-map chain for the front buffer and back buffer to be used as a two-level quadtree representation. However, we seek a method which takes advantage of separability as in [dL07]. We achieve this using, what we refer to as, the *pseudo-separable cross bilateral filter* (PSCBF). This involves performing CBF in two full-screen 1D passes as if it were separable. The method is used in [BSD08] for denoising, in screen space ambient occlusion, while preserving details near discontinuities in the depth buffer.

We will in the following show how convolution over a visible surface, at  $x_o$ , by a spatial Gaussian

$$G(\sigma, r) = \frac{1}{2\pi\sigma^2} e^{-\frac{r^2}{2\sigma^2}}$$

can be approximated using the cross bilateral filter. Let the distance in the  $XY$ -plane between a point  $x$  and  $x_o$  be denoted  $r_{12}(x)$  and the distance along the  $Z$ -axis is  $r_3(x)$ .

$$\begin{aligned} r_{12}(x) &= \sqrt{(x_o - x)_1^2 + (x_o - x)_2^2} \\ r_3(x) &= |(x_o - x)_3| \\ \|x_o - x\| &= \sqrt{r_{12}(x)^2 + r_3(x)^2} \end{aligned}$$

Furthermore, let the irradiance be given by the function  $I(x)$  and let  $S_{x_o}$  be a local patch of surface which is visible to the observer. Thus we perform substitution such that integration is done over the  $XY$ -plane,  $p \in \mathbb{R}^2$ , placed one unit from the observer.

$$\begin{aligned} \int_{S_{x_o}} G(\sigma, \|x_o - x\|) I(x) dx &= \int_{S_{x_o}} G(\sigma, r_{12}(x)) e^{-\frac{r_3(x)^2}{2\sigma^2}} I(x) dx \\ &= \int_{\mathbb{R}^2} G(\sigma, r_{12}(x(p))) e^{-\frac{r_3(x(p))^2}{2\sigma^2}} I(x(p)) dx(p) \quad (1) \end{aligned}$$

In this substitution  $x(p)$  represents the first intersection along a ray from the observer at  $(0, 0, 0)$  along the direction  $(p_1, p_2, -1)$ . The differential  $dx(p)$  is the surface area subtended by  $dp$  seen from the observer as shown in figure 1. There exists a relation between the two differentials. Let  $\vec{\omega}$  be the unit vector from the surface point  $x$  to the observer at origo. Next,  $\phi_i$  is the angle between

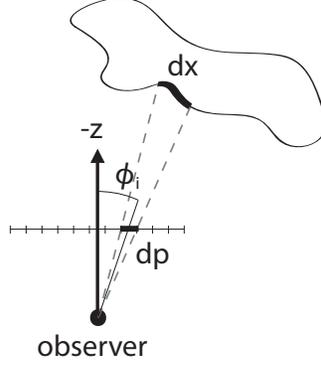


Figure 1: In this figure the relation between the differentials  $dx$  and  $dp$  is illustrated.

the  $Z$ -axis and  $-\vec{\omega}$  and  $\phi_j$  is the angle between the surface normal  $\vec{n}(x)$  and  $\vec{\omega}$ . Thus the relation is given by

$$w(p) = \cos^3(\phi_i) \frac{\|x(p)\|^2}{\cos(\phi_j)} \quad (2)$$

$$dx = w(p)dp \quad (3)$$

The equation (2) does not depend on the center,  $x_o = x(p_o)$ , of convolution. Subsequently, this may be premultiplied onto the irradiance  $I(x)$  prior to convolution. Let the coordinate, of  $x(p)$ , along the  $Z$ -axis be given by  $E : \mathbb{R}^2 \rightarrow \mathbb{R}$ . Finally, we use the approximation that there exists a linear **local** relation between  $r_{12}(x)$  and  $\|p - p_o\|$ . In summary,

$$\tilde{I}(p) = I(x(p)) \cdot w(p) \quad (4)$$

$$r_3(x) = |E(p) - E(p_o)| \quad (5)$$

$$r_{12}(x(p)) \simeq K_{p_o} \cdot \|p - p_o\| \quad (6)$$

we use equations (4)–(6) in equation (1) and obtain the following

$$\begin{aligned} \int_{S_{x_o}} G(\sigma, \|x_o - x\|) I(x) dx &\simeq \int_{\mathbb{R}^2} G(\sigma, K_{p_o} \cdot \|p - p_o\|) e^{\frac{-|E(p) - E(p_o)|^2}{2\sigma^2}} \tilde{I}(p) dp \\ &= \frac{2\pi\sigma^2}{K_{p_o}^2} \int_{\mathbb{R}^2} G\left(\frac{\sigma}{K_{p_o}}, \|p - p_o\|\right) G(\sigma, |E(p) - E(p_o)|) \tilde{I}(p) dp \\ &= \frac{2\pi\sigma^2}{K_{p_o}^2} CBF[\tilde{I}, E](p_o) \end{aligned} \quad (7)$$

where  $CBF$  represents an **unnormalized** cross bilateral filter applied to  $\tilde{I}$  and  $E$  using the variance  $\frac{\sigma}{K_{p_o}}$  and  $\sigma$  respectively. To arrive at this result we have

used two approximations. The first is that the surface  $S_{x_o}$  is all over visible, seen from the observer, and the second is approximation (6). These will be discussed in the next section.

## 4 Overlapping Surfaces and Filter Size

To perform the integration in screen space for an arbitrary surface  $S_{x_o}$  we have to use *depth peeling* [Mam89] which requires rendering to multiple buffers. In [vL07] only two layers are used. The first contains the frontmost surfaces and the second the backmost. This presents a problem when two objects on the screen overlap. For this reason the method in [vL07] renders every material with subsurface scattering into a texture. However, this solution only works, correctly, for materials of a convex shape.

For our solution we prefer to render all materials directly to the draw buffer. Ideally, multiple layers (more than two) are necessary for concave objects. This strategy has been used, for instance, to achieve global illumination [RGS09]. Thus, summation over all layers is necessary in equation (7). However, the convolution is optimized by assuming that only contributions within some distance  $r_{max}$ , of the center  $x_o$ , need to be accumulated during convolution. The reflectance profile is monotonically decreasing which agrees with this approximation. Thus, the domain of convolution is the 2D bounding region  $d$  of the projected sphere as indicated in figure 2.

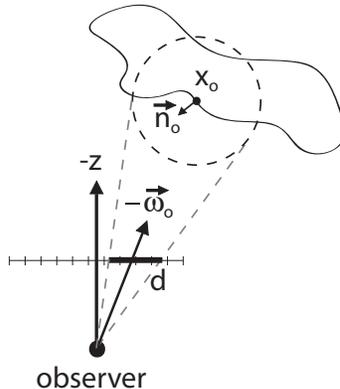


Figure 2: The filter kernel, of size  $d$ , is determined per pixel by a projection of the bounding sphere of radius  $r_{max}$ .

An approximate linear relation (6) between distance in screen-space and distance across the surface, in the  $XY$ -plane, is determined by assuming that the surface  $S_{x_o}$  is locally planar. Note that the dipole approximation based reflectance profile [PCW89] is derived based on the premise that the surface is a

plane-parallel medium. Thus, using the reflectance profile to integrate across an arbitrary surface implies this is already an assumed approximation in [JMLH01].

Let  $\theta_o$  be the angle between the  $Z$ -axis and  $\vec{n}_o = \vec{n}(x_o)$  and let  $\phi_o$  be the angle between  $\vec{\omega}_o$  and  $\vec{n}_o$ . It follows that

$$K_{p_o} = \frac{2 \cdot r_{max} \cdot |\cos \theta_o|}{d \cdot \cos \phi_o}$$

which is used in equation (7). The bound  $d$  and both trigonometric terms are determined in the  $XZ$ -plane and the  $YZ$ -plane separately. Thus the first Gaussian in equation (7) has an axis dependent variance. To obtain the trigonometric terms we use the surface normal of the primitive which is given by the interpolated vertex normal.

## 5 Optimizing for Skin

All current real-time methods separate subsurface scattering into two concepts: local subsurface scattering and transmitted light. Papers such as [JMLH01] and [DJ05], which are not real-time, integrate across the surface and accumulate without making such a distinction. The motivation for this separation is that local subsurface scattering accounts for most of the effect associated with human skin. Furthermore, it simplifies the approximation.

Local subsurface scattering is the approximation that subsurface scattering occurs within a surface region which is local to the center of convolution  $x_o$ . This region is contained within a disc and is nearly tangent to  $n_o$ . The disc has radius  $r_{max}$  and the reflectance profile is normalized over the disc. The approximation implies that the length of the shortest path across the surface from  $x$  to  $x_o$  is approximately the same as  $\|x - x_o\|$ . This is what makes simulation in a texture unwrap possible in papers such as [BL03] and [dL07].

The disc has roughly the same projected 2D bound  $d$  as the sphere when the tangent plane faces the observer. However, as  $x_o$  approaches the silhouette the bound approaches zero. Thus, we only need a single layer to simulate local subsurface scattering in screen-space. We use scalar projection to correct the bound  $d' = d \cdot \cos \phi_o$  for the disc. Again, the terms involved are determined in the  $XZ$ -plane and the  $YZ$ -plane separately.

Knowing that the analytical reflectance profile, or in our case the Gaussian, integrates to a constant over the disc  $D_{x_o}$  allows us to importance sample and to adjust a relatively rough discrete evaluation. Anything outside the domain of the disc is considered irrelevant. A weight is associated with each sample and the sum of all weights is one. The radius of significant contributions  $r_{max}$  is roughly  $8mm$  for human skin (see [dL07]). For such a small neighborhood relative to the distance to the observer, equation (2) is approximately constant  $w(p_o) \simeq w(p)$  within the disc. Thus, instead of using equation (4) we convolve against the irradiance  $I(x)$ . We may do this using the **normalized** cross bilateral filter **CBF** since the left side of approximation (7) is one when  $I(x) = 1$  for  $x \in D_{x_o}$ .

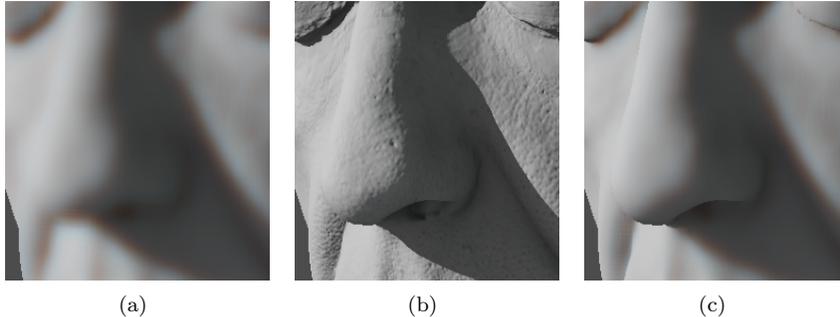


Figure 3: A regular 2D blur of figure 3(b) is shown in figure 3(a). The corresponding result using approximation (8) gives the result in figure 3(c).

From this it follows that

$$\int_{D_{x_o}} G(\sigma, \|x_o - x\|) I(x) dx \simeq \mathbf{CBF}[I, E](p_o) \quad (8)$$

where the filtering region is determined by  $d'$ . In figure 3(a) it is shown how a regular 2D Gaussian filter blurs out the silhouette of the nose. In contrast, figure 3(c) shows how our implementation of approximation (8), with the variable filter size  $d'$ , preserves features of the face. The same standard deviations  $\sigma_r$ ,  $\sigma_g$  and  $\sigma_b$  were used to produce the two figures.

To achieve plausible results we find that it is adequate to approximate the filter by using two 1D cross bilateral filters (horizontal and vertical). Note that as the surface aligns with the  $XY$ -plane the filter is reduced to a regular spatial Gaussian which is separable. Furthermore, the diffusion approximation is based on only two bands of spherical harmonics coefficients to represent the diffuse radiance and the dipole based reflectance profile is derived based on additional rough approximations. Thus numerical accuracy is largely compromised.

To perform the 1D convolution we represent the Gaussian by a piecewise constant approximation using a fixed number of  $N$  taps. These are distributed such that the cumulative distribution function yields an even distribution of probabilities,  $\frac{1}{N}$ , according to the variance  $\sigma_r$  associated with the red channel. This is because the reflectance profile for the wavelength associated with red is broader than that of green and blue. Corrected weights/probabilities are used for green and blue to use the same texture reads for these. For sampling the frame buffer we disable hardware filtering since adjacent pixels can represent locations which are far apart.

Since the weights are predetermined we do not need to compute  $K_{p_o}$  to perform the filter. We can proportionally distribute the taps across the coverage  $d'$ . An issue which we have not yet addressed is pixels  $p$  which do not represent skin. We identify these, similar to [DS03], using an alpha channel which is one

for skin and zero otherwise. Thus such pixels are identified and subsequently discarded during convolution. This also implies that convolution is only performed if alpha at the center  $x_o$  of convolution is non zero. Using stencil is an efficient way to achieve this.

## 6 Transmittance

It is difficult to support light transmitted through the medium once the approximation of local subsurface scattering is used in the applied model such as in [dL07] and our method in section 5. For an arbitrary surface there is no clear distinction between locations which are local to  $x_o$  and those which transmit  $x_t$ . As pointed out in [dL07] the result becomes too bright at locations where  $x_o$  and  $x_t$  are too close at the surface. We accumulate transmitted light similar to how it is done in [dL07] but require that the tangent planes at  $x_o$  and  $x_t$  are opposing. Furthermore, transmitted contributions are penalized using a scale by the term  $\sqrt{\max(0, -\vec{n}_t \bullet \vec{n}_o)}$ .

We render out back-facing primitives, in screen-space, as is done in [vL07]. However, the ears of a human head are concave and the surface tends to curl. Thus a ray through the ear can intersect multiple back-facing surfaces. For the subtle effect of glowy ears and nostrils we choose a more simple and practical solution to solve this problem without resorting to additional layers. We use a low resolution texture to mark the nose and the back of the ears. These are then rendered into the back-buffer which ensures we, subjectively, obtain the most important transmitters for a human head. Alternatively, one might implement a more general and brute force method to search for the transmitter with the brightest contribution. Another option is to use a translucent shadow map as in [DS03] or use a shadow map to determine the transmitting location as in [dL07]. However, both of these methods assume a point/directional light source. Instead of this we simply integrate in the back-buffer, within  $d'$ , using 16 taps.

## 7 Results

For testing we use a contribution from the company XYZRGB. This is a low polygonal model with a normal map which represents a high resolution scan of Roberto.

The reflectance profile associated with human skin exhibits exponential decay. As pointed out in [BL03], it is reminiscent of a spike and a broad base. It is possible to use our solution, specified by equation (8), to convolve against the specific reflectance profile given in [dL07] which consists of a sum of six Gaussians. However, in [LM07] the irradiance prior to convolution is used, as an approximation, to represent the spike. The remainder of the profile is thus the low frequency broad base. As opposed to six Gaussians we use a profile similar to that of [LM07]. However, in our case the broad base is represented

by a single Gaussian per channel. We have empirically chosen the parameters in table 1 for our reflectance profile and for local subsurface scattering we use the radius  $r_{max} = 4.5mm$  to compute the filtering bound  $d'$ . The values, for the variance, in table 1 are given relative to units in  $mm$ .

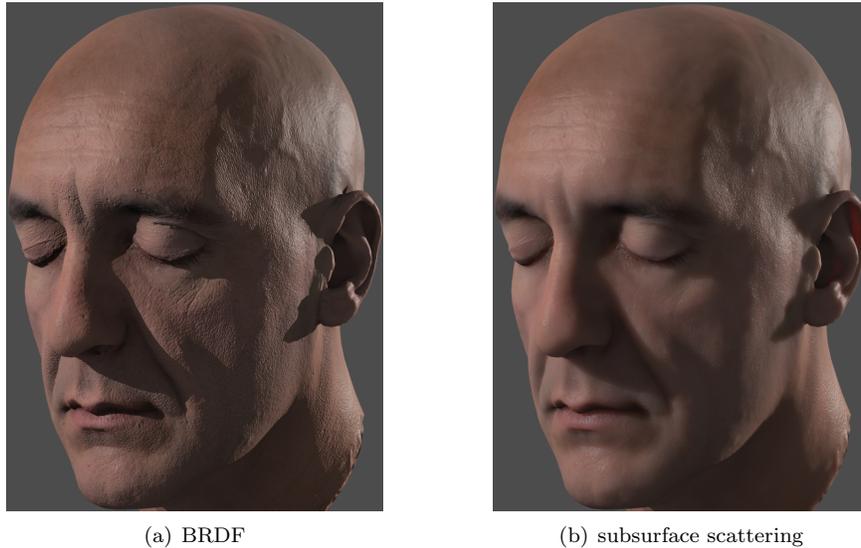


Figure 4: This figure shows the difference between using our method for simulation of subsurface scattering and disabling it (BRDF). The softness of skin and red fleshy tones emerge when used as seen in figure 4(b).

For our implementation of the filter **CBF** we use  $N = 13$  taps along each axis. Results without simulation of subsurface scattering are shown in figure 4(a). The appearance seems hard and lifeless. In contrast with simulation the

	Red	Green	Blue
Spike	15%	18%	20%
Variance	2.3	1.4	1.1

Table 1: Parameters of the reflectance profile.

appearance of the skin is softer and we see subtle red fleshy tones occurring in the shadows, wrinkles and pores. Furthermore, we see light transmitted through the ear but also the true rough terrain of human skin revealed by the specular reflection on the side of the head. A close-up is given in figure 5 to show the effect.

Given a fixed amount of taps our method scales, roughly, linearly by the

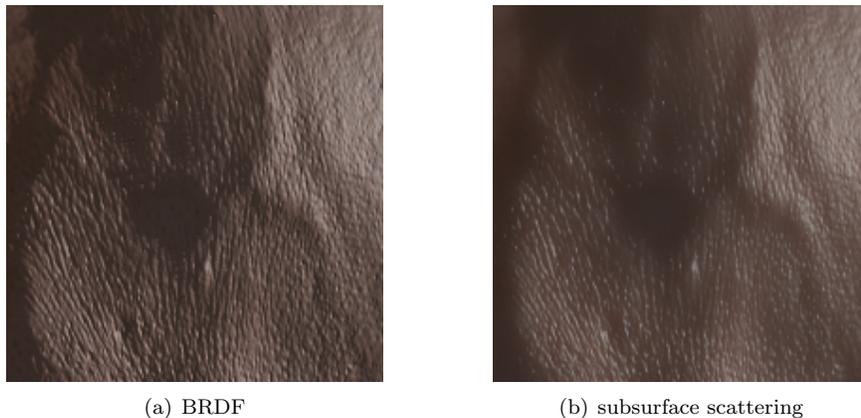


Figure 5: In figure 5(b) we see a close-up of the effect of our method at the side of the head of Roberto. The appearance when disabled is shown in figure 5(a) and is that of a hard material.

amount of pixels on the screen which represent skin.

We implemented a second version which uses horizontal (and subsequently vertical) mip maps for filtering. These were custom built to take depth discontinuities into account. Finally, filtering was done manually in the shader. We have omitted the results since they were roughly the same and execution was significantly slower. For future work it would be interesting to compare results for translucent objects where  $r_{max}$  is greater, i.e., a few centimeters.

## 8 Conclusion

We have shown how the appearance of skin can be approximated in screen-space using the cross bilateral filter with the filter radius chosen per pixel. More specifically we have derived an approximation for a local convolution, by a Gaussian, across the surface of a mesh.

We based our results on the assumption that the spike and the broad base of the reflectance profile can be approximated by the original image and a Gaussian convolution, across the surface, respectively. However, our technique can also be applied to the sum of Gaussians model used by Eugene d'Eon et al. in [dL07] but as in their work this requires multiple convolutions.

The most significant features of our method is that: we use pseudo-separability, to accelerate the process, and that the simulation is executed once per frame regardless of the amount of visible characters or light sources.

**Acknowledgments.** This author would like to thank John Hable for helpful discussions and Swaminathan Narayanan for constructive comments and proof reading. Finally, thank you to XYZRGB for allowing their scan of Roberto to be used for testing.

## References

- [BL03] George Borshukov and J. P. Lewis. Realistic human face rendering for "the matrix reloaded". In *SIGGRAPH '03: ACM SIGGRAPH 2003 Sketches & Applications*, pages 1–1, New York, NY, USA, 2003. ACM.
- [BSD08] Louis Bavoil, Miguel Sainz, and Rouslan Dimitrov. Image-space horizon-based ambient occlusion. In *SIGGRAPH '08: ACM SIGGRAPH 2008 talks*, pages 1–1, New York, NY, USA, 2008. ACM.
- [DJ05] Craig Donner and Henrik Wann Jensen. Light diffusion in multi-layered translucent materials. *ACM Trans. Graph.*, 24(3):1032–1039, 2005.
- [dL07] Eugene d'Eon and David Luebke. *GPU Gems 3*, chapter 14, pages 293–347. Addison-Wesley Publishing, 2007.
- [DS03] Carsten Dachsbacher and Marc Stamminger. Translucent shadow maps. In *EGRW '03: Proceedings of the 14th Eurographics workshop on Rendering*, pages 197–201, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.
- [JB02] Henrik Wann Jensen and Juan Buhler. A rapid hierarchical rendering technique for translucent materials. *ACM Trans. Graph.*, 21(3):576–581, 2002.
- [JMLH01] Henrik Wann Jensen, Stephen R. Marschner, Marc Levoy, and Pat Hanrahan. A practical model for subsurface light transport. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 511–518, New York, NY, USA, 2001. ACM.
- [JSG09] Jorge Jimenez, Veronica Sundstedt, and Diego Gutierrez. Screen-space perceptual rendering of human skin. *ACM Transactions on Applied Perception*, 6(4), 2009.
- [LM07] Anders Langlands and Tom Mertens. Noise-free bssrdf rendering on the cheap. In *SIGGRAPH '07: ACM SIGGRAPH 2007 posters*, page 182, New York, NY, USA, 2007. ACM.
- [Mam89] Abraham Mammen. Transparency and antialiasing algorithms implemented with the virtual pixel maps technique. *IEEE Comput. Graph. Appl.*, 9(4):43–55, 1989.

- [PCW89] M. Patterson, B. Chance, and B. C. Wilson. Time resolved reflectance and transmittance for the non-invasive measurement of tissue optical properties. *Appl. Opt.*, 28:2331–2336, 1989.
- [RGS09] Tobias Ritschel, Thorsten Grosch, and Hans-Peter Seidel. Approximating dynamic global illumination in image space. In *I3D '09: Proceedings of the 2009 symposium on Interactive 3D graphics and games*, pages 75–82, New York, NY, USA, 2009. ACM.
- [vL07] Brecht van Lommel. Subsurface scattering. Available at <http://www.blender.org/development/release-logs/blender-244/subsurface-scattering/>, 2007.