# A Practical Analytic Model for the Radiosity of Translucent Scenes

Yu Sheng*[1], Yulong Shi[2], Lili Wang[2], and Srinivasa G. Narasimhan[1]

[1]The Robotics Institute, Carnegie Mellon University
[2]State Key Laboratory of Virtual Reality Technology and Systems, Beihang University

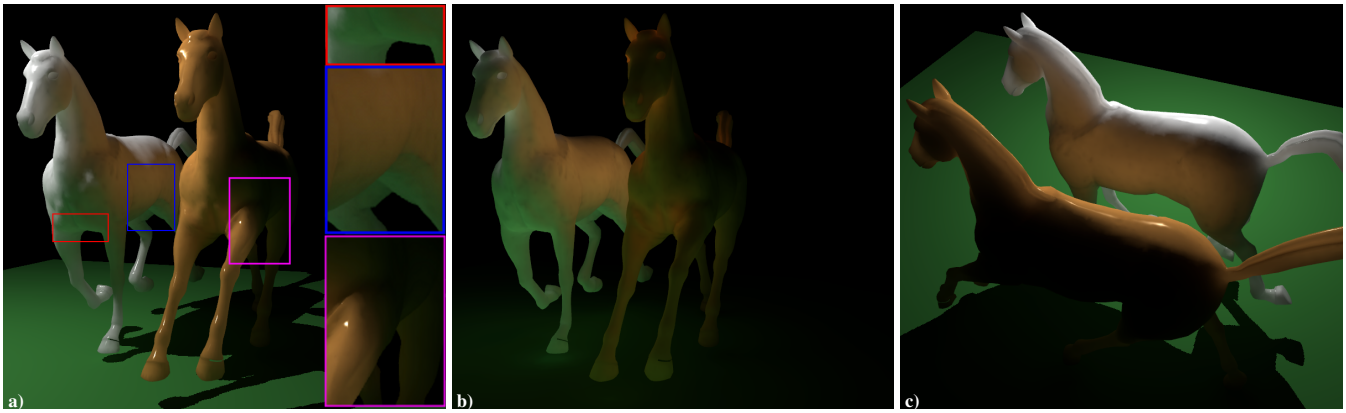


**Figure 1:** *Inter-reflection and subsurface scattering are closely intertwined for scenes with translucent objects. The main contribution of this work is an analytic model of combining diffuse inter-reflection and subsurface scattering (see Figure 2). One bounce of specularities are added in a separate pass. a) Two translucent horses (63k polygons) illuminated by a point light source. The three zoomed-in regions show that our method can capture both global illumination effects. b) The missing light transport component if only subsurface scattering is simulated. c) The same mesh rendered with a different lighting and viewing position. Our model supports interactive rendering of moving camera, scene relighting, and changing translucencies.*

## Abstract

Light propagation in scenes with translucent objects is hard to model efficiently for interactive applications. The inter-reflections between objects and their environments and the subsurface scattering through the materials intertwine to produce visual effects like color bleeding, light glows and soft shading. Monte-Carlo based approaches have demonstrated impressive results but are computationally expensive, and faster approaches model either only inter-reflections or only subsurface scattering. In this paper, we present a simple analytic model that combines diffuse inter-reflections and isotropic subsurface scattering. Our approach extends the classical work in radiosity by including a subsurface scattering matrix that operates in conjunction with the traditional form-factor matrix. This subsurface scattering matrix can be constructed using analytic, measurement-based or simulation-based models and can capture both homogeneous and heterogeneous translucencies. Using a fast iterative solution to radiosity, we demonstrate scene relighting and dynamically varying object translucencies at near interactive rates.

**CR Categories:** I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Radiosity.

**Keywords:** radiosity, subsurface scattering, inter-reflection

*{shengyu, srinivas}@cs.cmu.edu, {shiyl,lily_w}@vrlab.buaa.edu.cn

## 1 Introduction

Accurate rendering of translucent materials such as leaves, flowers, marble, wax, and skin can greatly enhance realism. The interactions of light within translucent objects and in between the objects and their environments produce pleasing visual effects like color bleeding (Figure 1), light glows and soft shading. The two main mechanisms of light transport — (a) scattering beneath the surface of the materials and (b) inter-reflection between surface locations in free space — combine in complex ways (see Figure 2) making it challenging to render such scenes quickly and realistically.

Monte-Carlo ray tracing [Jensen et al. 1999; Pharr and Hanrahan 2000] can be used to faithfully render both inter-reflection and subsurface scattering but is too slow to be practical. Jensen and Buhler [2002] models the inter-reflection between the diffuse surface and a translucent object by substituting translucency inter-reflection with diffuse inter-reflection. Computational speedups have been achieved by combining Photon Mapping with analytical models for subsurface scattering to capture interesting global illumination effects, such as inter-scattering and caustics, for translucent materials [Donner and Jensen 2007]. However, this method does not support multi-layered objects and heterogeneous materials. Furthermore, none of them achieve interactive rendering speed. Precomputed radiance transfer (PRT) methods have achieved interactive rendering and relighting of translucent objects by extensive precomputation, usually in hours [Sloan et al. 2002; Sloan et al. 2003; Wang et al. 2005; Wang et al. 2008b]. Unfortunately, all these approaches focus on rendering a single translucent object. Therefore, interactive rendering of translucent objects with both inter-reflection and subsurface scattering remains a challenge.

In this paper, we present a simple analytic model of the light transport between translucent objects and diffuse environment. We assume that the translucent objects are highly scattering and hence the reflected and transmitted light is diffuse. This is valid for a

wide range of materials and is commonly used in rendering and acquisition systems [Jensen et al. 2001; Goesele et al. 2004; Wang et al. 2008a]. We also assume that any other objects in the scene are diffuse. In this setting, the appearance of the scene can be computed by the radiosity at every surface location. Our model extends the classical work in radiosity [Goral et al. 1984] by including a subsurface scattering matrix that operates in conjunction with the traditional form-factor matrix. The subsurface scattering matrix can be constructed using any analytical model (dipole [Jensen et al. 2001] and multi-pole [Donner and Jensen 2005]) or using simulations [Wang et al. 2008a] or measurements [Goesele et al. 2004]. Hence, our method can render both homogeneous and heterogeneous materials.

Being a radiosity-like approach, the method requires a one-time precomputation (10 min) and storage (1.5 GB for a 40k polygon mesh) of the form-factor and subsurface scattering matrices. But once this is done, rendering is achieved at interactive rates using our GPU implementation of a standard iterative solver. This allows us to perform fast relighting (5-10 fps) of scenes under different types of sources (spot-light and environment maps) with arbitrary radiance distributions. Further, our parametrization of the subsurface scattering can be exploited to vary the translucency of objects at near interactive rates (1-4 fps). If we wish to perform only relighting, we can precompute the inverse. Each relighting computation thus requires only one matrix-vector product computation and we achieve 50-80 fps. We demonstrate results on objects with complex shapes (including multiple-layered objects like a flower (Figure 9)) and showcase effects like color bleeding, glows, and object-environment interactions (Figure 1).

## 2 Background and Related Work

**Radiosity** is a classical global illumination technique to render diffuse environments [Goral et al. 1984]. The entire scene is discretized into surface patches. An important part of the formulation is the form-factor matrix which represents the fraction of energy leaving one surface patch and arriving directly at another. The final rendering can be computed by solving a linear system,

$$\mathbf{B} = (\mathbf{I} - \mathbf{F})^{-1}\mathbf{L} \tag{1}$$

where, $\mathbf{B}$ is the vector of radiosities of all patches, $\mathbf{I}$ is an $n \times n$ identity matrix, $\mathbf{F}$ is the form-factor matrix, and $\mathbf{L}$ is the vector of initial emittances of all patches. The form-factor matrix needs to be precomputed and can be expensive for large scenes. Different strategies have been proposed to accelerate the original algorithm, such as progressive radiosity [Cohen et al. 1988] and hierarchical radiosity [Hanrahan et al. 1991]. Radiosity performs well for materials with a matte appearance. Specular and glossy materials require ray tracing based approaches to create realistic renderings [Wallace et al. 1987; Christensen et al. 1997]. Radiosity has also been extended to compute the light transport in the context of participating media by introducing the volume-volume and volume-surface form factors [Rushmeier and Torrance 1987]. However, light transport due to inter-reflection and volumetric scattering is tangled together in the formulation. Therefore, with the zonal method, once the properties of participating media are changed, all the form-factors need to be recomputed. In contrast, our method models these two light transport separately and supports translucency changes at near-interactive speed. Rushmeier et al. [1990] have also extended radiosity for translucent surfaces, but only limited to the transmission from the opposite side of the same patch.

**Subsurface scattering** can be described by an 8-D Bi-directional distribution function (BSSRDF [Nicodemus et al. 1977]),

$$s(x_i, \vec{\omega}_i; x_o, \vec{\omega}_o) = \frac{dL_o(x_o, \vec{\omega}_o)}{d\Phi_i(x_i, \vec{\omega}_i)}. \tag{2}$$

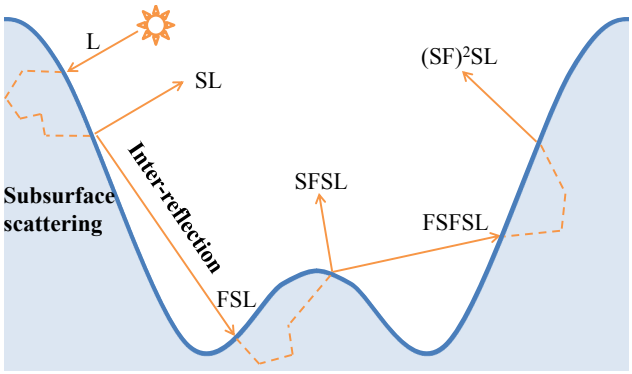


**Figure 2:** *Light transport in the presence of a non-convex translucent object. Light incident at a surface patch is scattered beneath the surface of the object and exits and re-enters the object at a different location. The propagation of light in free space is termed as inter-reflection and within the object as subsurface scattering. The radiosity of the object is the result of a series of such light interactions and is captured by our model.*

$L_o(x_o, \vec{\omega}_o)$ is the outgoing radiance at a surface point $x_o$ in direction $\vec{\omega}_o$, and $\Phi_i(x_i, \vec{\omega}_i)$ is the incoming radiance at a surface point $x_i$ in direction $\vec{\omega}_i$. A brute-force evaluation of this high-dimensional function using Monte-Carlo simulations [Pharr and Hanrahan 2000], is very expensive and impractical to use in most applications. Dipole diffusion model [Jensen et al. 2001] for highly scattering materials is easy to implement, much faster than Monte-Carlo simulations and has been widely used in computer graphics. Later works extended this diffusion model to support multi-layer translucent materials [Donner and Jensen 2005], heterogeneous translucent materials [Wang et al. 2008a; Arbree et al. 2011], anisotropic diffusion [Jakob et al. 2010]. To accelerate the rendering speed of translucent objects, hierarchical methods [Jensen and Buhler 2002; Arbree et al. 2008] have been proposed to support large scenes with complex illumination.

For highly scattering materials, the angular dependency of incoming and outgoing directions $\vec{\omega}_i$ and $\vec{\omega}_o$ can be removed [Jensen et al. 2001]. With this assumption, [Lensch et al. 2002] and [Carr et al. 2003] proposed to render translucent objects using radiosity to achieve real-time frame rates. Hierarchical extensions of the two methods managed to interactively relight, edit materials, and change geometry for moderately complex translucent objects [Mertens et al. 2003]. However, all these approaches focus only on light scattering beneath the object's surface (dotted lines in Figure 2) and assume that light does not re-enter the object once it exits the object. Our work builds on these approaches and models all the light interactions shown at Figure 2.

**Precomputed Radiance Transfer** provides a possible solution for real-time rendering by precomputing the transfer vectors represented with spherical harmonics [Sloan et al. 2002] or wavelet basis [Ng et al. 2003]. At run-time, relighting of each vertex reduces to a vector inner-product and thus achieves real-time rates. It has been extended to render translucent objects with both single and multiple scattering components at interactive rates [Wang et al. 2005] or even real-time speed with only multiple scattering [Wang et al. 2008b]. However, to accurately capture the light propagation between translucent objects and their environment, PRT-based methods require precomputation of global illumination under many (9-25) lighting bases [Sloan et al. 2002], each of which can take several hours if using a Monte-Carlo variant. Furthermore, most PRT methods focus on distant lighting for efficient compression of incident light. Including near-field lighting dramatically increases the precomputation time as even more basis functions are needed.

In contrast, our method does not precompute the light transfer matrix, but the form-factor and subsurface scattering matrices, which takes only a few minutes (vs. hours/days for PRT) and works for both distant and near lighting.

## 3 Analytically combining Inter-reflection and Subsurface Scattering

Figure 2 shows the order of light transport through a non-convex and translucent object. The propagation of light in free space is termed inter-reflection and the propagation within the object is termed subsurface scattering (dotted rays). We are interested in modeling the potentially infinite series of such light interactions (bounces) to compute the final radiosity of the object.

**First bounce:** The scene is discretized into $n$ surface patches. The portion of light that is scattered from patch $i$ to patch $j$ is denoted by $\mathbf{S}_{ji}$. If $\mathbf{L}_i$ is the irradiance at patch $i$, then the light received by patch $j$ is defined as the *local scattering* or as the first bounce:

$$\mathbf{S}_{ji} \cdot \mathbf{L}_i \,. \tag{3}$$

$S_{ji}$ is called the *scattering factor*, which is the relative amount of the light transported from patch $i$ to $j$ due to subsurface scattering. Note that this term is called throughput factor by [Lensch et al. 2002] and [Carr et al. 2003]. If the areas of the patches $i$ and $j$ are $A_i$ and $A_j$, they write $\mathbf{S}_{ji}$ in a manner similar to the form-factor as:

$$\mathbf{S}_{ji} = \frac{1}{A_i} \int_{A_i} \int_{A_j} s(i,j) dA_j dA_i \tag{4}$$

where, $s(i,j)$ is the simplified BSSRDF function (Equation 2) that only depends on the incident and outgoing points.

**Second and $N$-th bounce:** The light that exits surface patch $j$ due to the first bounce is $\sum_i \mathbf{S_{ji}} \mathbf{L}_i$, which is then transported to other surface patches in the scene in free space. Suppose that $\mathbf{F}_{kj}$ is the form-factor between patches $j$ and $k$, the light received by patch $k$ from patch $i$ is $\mathbf{F}_{kj} \sum_i (\mathbf{S}_{ji} \mathbf{L}_i)$. We obtain the total light received by patch $k$ by adding contributions from all patches $j$ as $\sum_j \mathbf{F}_{kj} \sum_i (\mathbf{S}_{ji} \mathbf{L}_i)$. In matrix form, we thus have:

$$\mathbf{FSL} \tag{5}$$

where, both $\mathbf{F}$ and $\mathbf{S}$ are $n \times n$ matrices. Similar to the analysis in Equation (3), the *local scattering* of this bounce is $\mathbf{SFSL}$. This term is the second bounce of light. In general, the $N$-th bounce of light can be written as:

$$(\mathbf{SF})^{N-1} \mathbf{SL}. \tag{6}$$

Then, the **entire light transport or radiosity** of the scene is computed as the sum of all bounces of light:

$$\mathbf{B} = \sum_{N=1}^{\infty} (\mathbf{SF})^{N-1} \mathbf{SL} = (\mathbf{I} - \mathbf{SF})^{-1} \mathbf{SL} \tag{7}$$

where, $\mathbf{I}$ is the $n \times n$ identity matrix. Note that this equation is similar to the classical radiosity equation (Equation 1) but incorporates the additional subsurface scattering matrix in a simple and elegant manner. In particular, if the entire scene is opaque, $\mathbf{S} = \mathbf{I}$ and we obtain Equation 1. Since light propagation in free-space (vacuum) is fundamentally different than propagation through materials, the matrices $\mathbf{F}$ and $\mathbf{S}$ have different characteristics. Treating them as separate entities of light transport allows us to control the translucency of the objects, for instance.

As in classical radiosity, the above equation can be efficiently solved by iterative methods, such as Jacobi or Gauss-Seidel. Since the norm of the form-factor matrix $\|\mathbf{F}\| < 1$, the radiosity equation is proven to converge if not all the surfaces are perfect reflectors (with albedo equal to 1) [Cohen and Wallace 1993]. According to the physical meaning of $\mathbf{S}_{ji}$, the sum of all the percentages of light that scatters from patch $i$ to others should be less than 1. That is, for any $i$, $\sum_j \mathbf{S}_{ji} < 1$, then

$$\|\mathbf{S}\|_1 = \max_{1 \le i \le n} \sum_j \mathbf{S}_{ji} < 1. \tag{8}$$

Therefore, the spectral radius $\rho$ of matrix $\mathbf{S}$ satisfies $\rho(\mathbf{S}) \le \|\mathbf{S}\|_1 < 1$. Since we also have $\rho(\mathbf{F}) \le \|\mathbf{F}\| < 1$, then $\rho(\mathbf{SF}) \le \rho(\mathbf{S})\rho(\mathbf{F}) < 1$. Therefore, Equation (7) always converges.

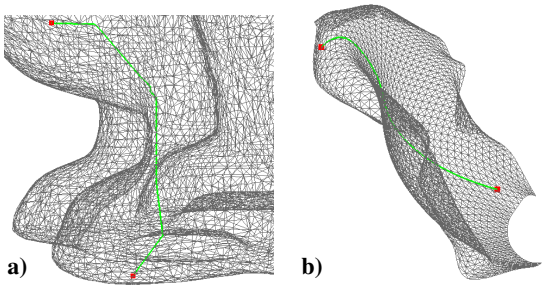


**Figure 3:** *The green line in the left image shows the optical path for two points on the bunny. The distance of the straight line that connects these two points will greatly over-estimate the scattering factor. The right image shows a geodesic path on a thin rose petal.*

## 4 Computing Subsurface Scattering Matrix

The model in Equation (7) provides a framework for rendering translucent objects and their interactions with diffuse environments. The subsurface scattering matrix $\mathbf{S}$ can be computed using either analytical models, numerical simulations or measurements. We describe this process below.

Let us start with the numerical evaluation of Equation (4) that computes the light throughput between two surface patches. In practice, since radiance decays (roughly) exponentially through materials, the number of samples of $s(i,j)$ used in the integral can be reduced as the distances between patches increase. Further, depending on the desired quality of rendering, a threshold can be set for the distance beyond which subsurface scattering can be safely ignored.

For **homogeneous materials**, the sampling kernel $s(i,j)$ simplifies to a 1-D function $s(r)$ that only depends on the distance $r$ between the sampling points on each surface. In this case, our model can take either analytical formulations such as dipole diffusion [Jensen et al. 2001], or 1-D scattering profiles obtained by measurement. A common approach to acquire $s(r)$ is to illuminate a planar slab of the sample material and measure the intensity distribution on both sides of the slab to obtain the reflection and transmission profiles.

However, when applying the dipole, multi-pole models or observed profiles to objects with arbitrary shapes, it is unclear how the reflectance and transmittance components change due to the local curvature and global shape of the object. Premoze et al. [2004] propose using the most probable paths to compute multiple scattering in participating media. We follow an approach that is similar in spirit and parametrize the profiles as a function of the shortest optical path through the translucent objects between the light incident and exiting points, as shown in Figure 3a. Note that this is not the only path for light propagation but only a parametrization. For a convex object, this path is simply the euclidean distance. However, for objects with concavities, euclidean distance does not accurately

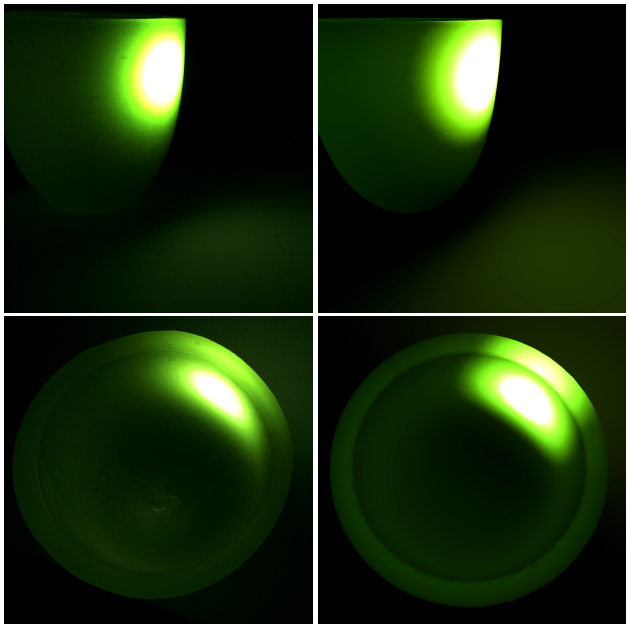

**Figure 4:** *Comparison between our renderings (right) and photographs (left) taken with a real wax bowl.*

capture how light propagates through the translucent objects. For example, in Figure 3a, part of the line segment that connects the two red points is outside the object. The light transport between different parts of the object through vacuum should be modeled by inter-reflection and not subsurface scattering. For objects with high curvature (for example, a long but narrow U-shape object), the difference between the length of actual optical path and euclidean distance can be significant.

To compute distance $r$, we construct a uniform voxel grid that covers the bounding box of the object. A ray-casting and surface-intersection test is performed to determine which voxels are within the object. Then, regarding each interior voxel as a node of a large graph, we determine the shortest path (and distances) between each interior voxel-pair [Dijkstra 1959]. For thin sheets like the petals of a flower (see Figure 3b), the shortest distance between light entry and exit locations is simply their geodesic distance [Mitchell et al. 1987]. This matrix of shortest distances needs to be computed only once and stored in memory for a particular scene. The subsurface scattering is then computed by just indexing, based on distances, into the reflection/transmission profiles. Thus, by simply re-computing the 1-D profiles, we can vary the translucency of the scene at near-interactive rates.

Since there is no constraint (other than energy conservation) the matrix $\mathbf{S}$ can represent an arbitrary **heterogeneous material**. Again, measurements such as those from the DISCO system [Goesele et al. 2004], or numerical evaluations [Wang et al. 2008a; Arbree et al. 2011], can be incorporated. We present a simple heuristic in the next section to generate plausible renderings by combining the scattering properties of multiple homogeneous materials.

## 5 Results

In this section, we demonstrate the efficiency and accuracy of our analytic model and algorithm. All our results were generated on a desktop with an Intel Quad-Core i7-2600k CPU and an NVIDIA Geforce GTX 590 graphics card. Our prototype implementation was tested on four scenes. Table 1 describes the number of polygons in each scene, memory, and the precomputation and rendering

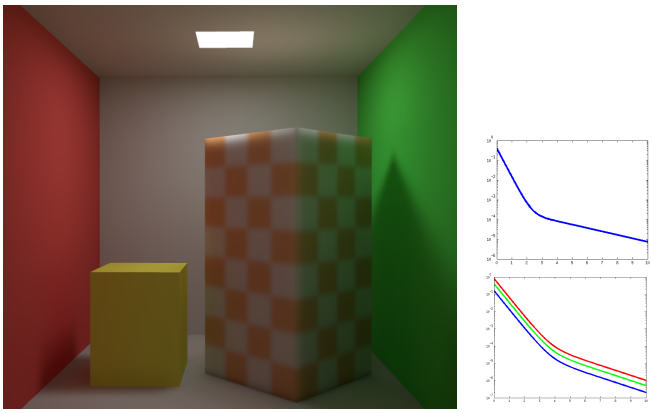

**Figure 7:** *Our model also supports heterogeneous translucent materials. The left image is the Cornell box scene with the tall block applied by a checkerboard pattern of translucencies. The right images are the scattering profiles of the white and orange squares. Note the color bleeding effects on both sides of the marble block and the orange color on the right part of the ceiling to the reflection from the translucent block.*

times. Please see supplementary material for more visualizations.

**Validation:** Radiosity is a well validated method and our model is a physics-based extension. The radiosity solution involves iterative solvers for a linear system, extensively studied for many years. In practice, for our test scenes, it takes 5-10 iterations to converge with an residual error of 0.5%. To further ensure the accuracy of our model, we compared our renderings of a **concave wax bowl** with photographs (Figure 4). The interior of the bowl is illuminated by a strong spot light (flashlight). The resulting brightness observed on the outside surface shows that the bowl is highly translucent. We setup a projector-camera system to measure the reflection and transmission profiles. We carefully calibrated the geometric relationship between the projector and the camera. The color and brightness of the projector were calibrated by a Gretag Macbeth® ColorChecker®. We measured the radially symmetric BSSRDF profiles by capturing the light distribution on the bowl while illuminating only one pixel of the projector. The colors of the source and intensity mapping (gamma) were done manually to match the photographs. Visual comparisons show that our renderings are very similar to the photos. Because we do not have an accurate mesh (with small bumps on the bowl), and spatial-varying scattering profiles, subtle differences in local shading can be observed between our renderings and the photos.

The **Cornell box** is a six-sided enclosure, with a yellow opaque box, and a tall marble block. The scattering profile of the marble block is computed using dipole approximation. Figure 5 demonstrates the different components of the light transport. Our method can correctly capture both inter-reflection and subsurface scattering. Note the red and green colors on sides of the translucent block due to the inter-reflection from colored walls.

As described in Section 4, the matrix of shortest distances is the most computational expensive part in computing the subsurface scattering. For a static scene, it needs to be computed only once and indexed into the 1D BSSRDF profiles to produce the subsurface scattering matrix. Figure 6 shows renderings when the translucency of the block is changed by varying the 1D BSSRDF profiles.

Figure 7 demonstrates that our model supports heterogeneous materials. Because we focus on modeling the interaction between subsurface scattering and inter-reflection, a sophisticated implementation of physically-accurate heterogeneous materials is beyond the

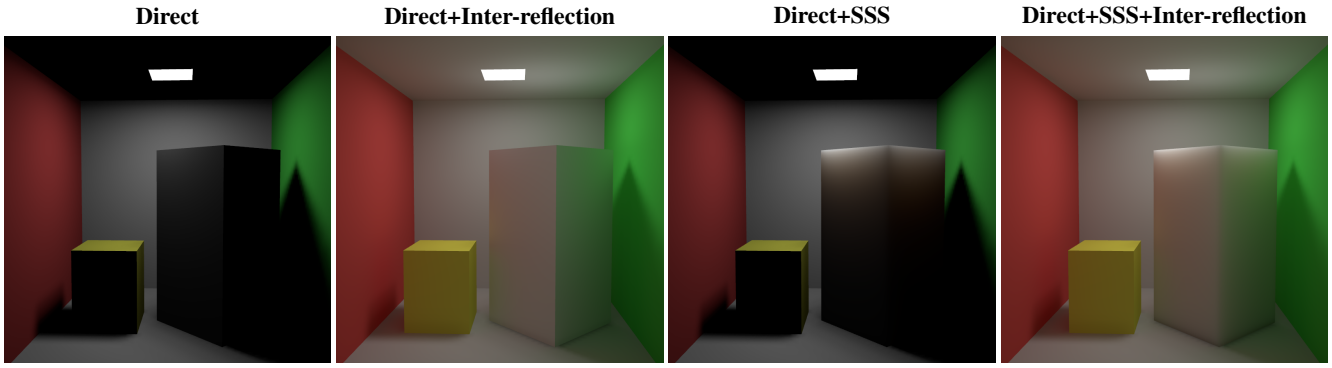


**Figure 5:** *Renderings of the Cornell box scene to show different components of light transport. From left to right, each image show the direct illumination, only diffuse inter-reflection, only subsurface scattering scattering, and the results with both global illumination effects. Our method can correctly capture the color bleeding on the translucent block due to the inter-reflection.*

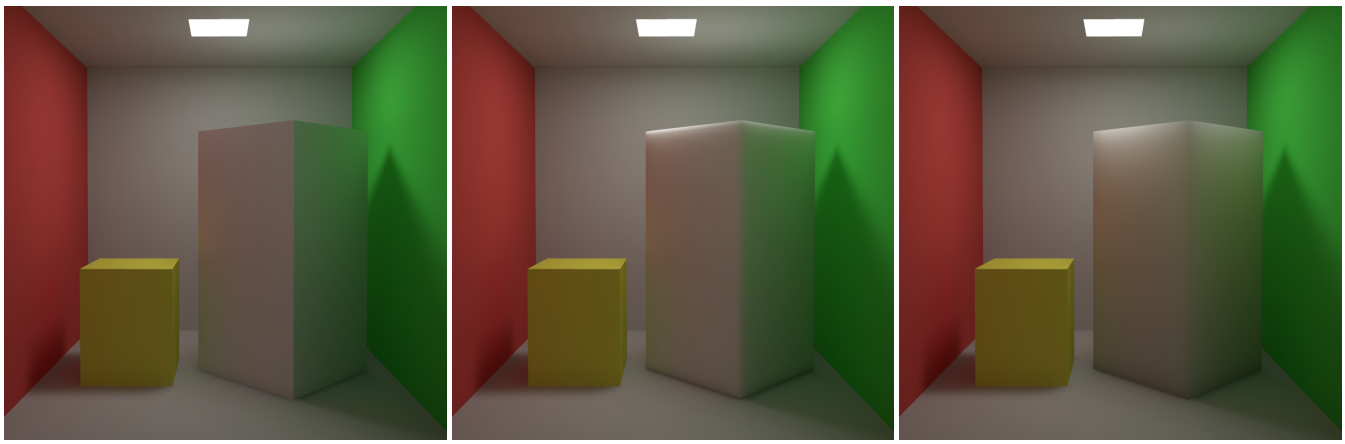

**Figure 6:** *The Cornell box scene with change in translucency. From left to right, the translucency of the objects increases. Note the color bleeding effects on the translucent block get less visible as the translucency increases.*

scope of this work. We instead implement a simple heuristic for heterogeneity that can capture effects on an object with patches of different homogeneous materials. We combine the scattering profiles of each materials according to how far light traverses in that material:

$$\left(S_1(0)\frac{r_1}{r_1+r_2}+S_2(0)\frac{r_2}{r_1+r_2}\right)\times\frac{S_1(r_1)S_2(r_2)}{S_1(0)S_2(0)} \qquad (9)$$

where, $S_1(r)$ and $S_2(r)$ are homogeneous scattering profiles, $S_1(0)$ and $S_2(0)$ are the values (diffuse albedo) at distance 0; $r_1$ and $r_2$ are the distances that light travels in each homogeneous block. Although *not physically accurate*, this heuristic can capture plausible spatially varying appearances.

The second scene consists of **three bowls and the Stanford Bunny** on a diffuse table. The bowls and the Bunny are translucent. The scattering profiles of the bowls are based on our measurement described above but we increased translucency and modulated the colors. Figure 8 captures the inter-reflections between different translucent objects. We add one bounce of specular reflection that directly reflected to the camera by translucent objects. Multiple bounce of specular reflection and more complex light paths that involve the mixing of specular and diffuse reflections are not rendered and we will set aside as future work. Figure 11 shows renderings of this scene with different translucencies. In the left image, the bowl is almost diffuse and the light inside does not spread out. Colors of the other two bowls are clearly visible on the body of the greenish bowl. As the translucency increases, the head of the bunny and the floor receive more light.

The third object is a **rose** whose petals are modeled as having two opposing faces (or sides). So, the number of polygons in the mesh is doubled. We simulate the light entering and exiting the faces on the same side by a reflectance profile and light entering and exiting on different sides by a transmittance profile. We assume constant thickness for the petals. For this result, we used parametrized curves that exponentially fall-off with distance as the scattering profiles ($S(r) = a_1 exp(-b_1 r) + a_2 exp(-b_2 r)$). Figure 9 shows the rendering as illuminated by the St. Peters Basilica environment map and a point light source. Notice how the light propagates through and between the petals. For comparison, the middle image shows the missing component if only subsurface scattering is simulated and the right image shows the component without the first light bounce if only inter-reflection is simulated.

The last scene is an animation with **two horses** galloping on a green floor, illuminated by a point light source. The entire animation contains 48 meshes (downloaded from MIT deformation database[1]). For each mesh, we precompute the form-factor and subsurface scattering matrices. Once the two matrices are loaded from hard drive , the computation is performed as other scenes. Figures 1 and 10 show renderings of different meshes produced by our algorithm. Please see the accompanied video for the full animation.

**Comparison with heuristic methods:** Our model is designed precisely to avoid different types of heuristics, for example, substituting translucency inter-reflection with diffuse inter-reflection [Jensen and Buhler 2002]. Depending on how diffuse

[1] http://people.csail.mit.edu/sumner/research/deftransfer/data.html

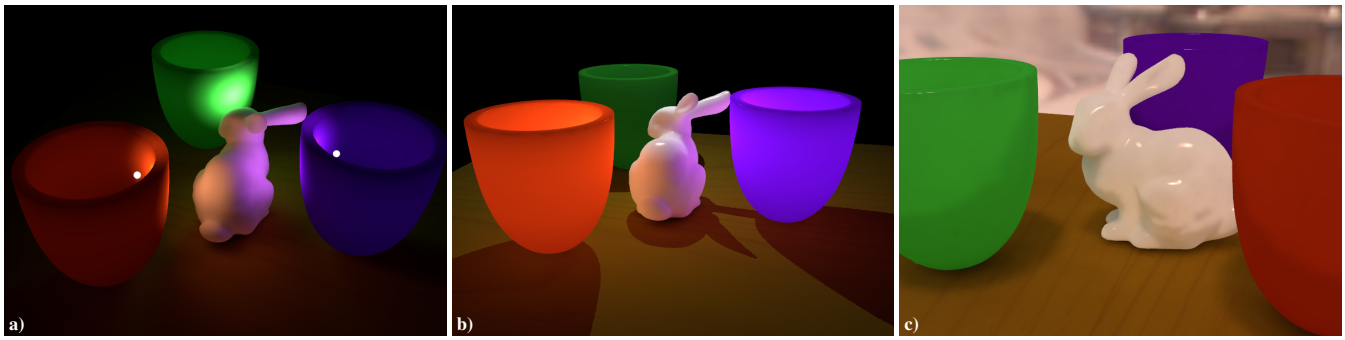


**Figure 8:** *Renderings of the "bowls and bunny" scene under different lighting conditions. a) Light scattered through three colored bowls illuminates different parts of the bunny. The scene is illuminated by three spot light sources, with one located inside each bowl. b) The same scene is illuminated by three point light sources, one above the green bowl and two inside the other two bowls. c) The same scene illuminated by environment lighting. We add one-bounce specular reflection on translucent objects for b) and c). Multiple bounces are not simulated.*

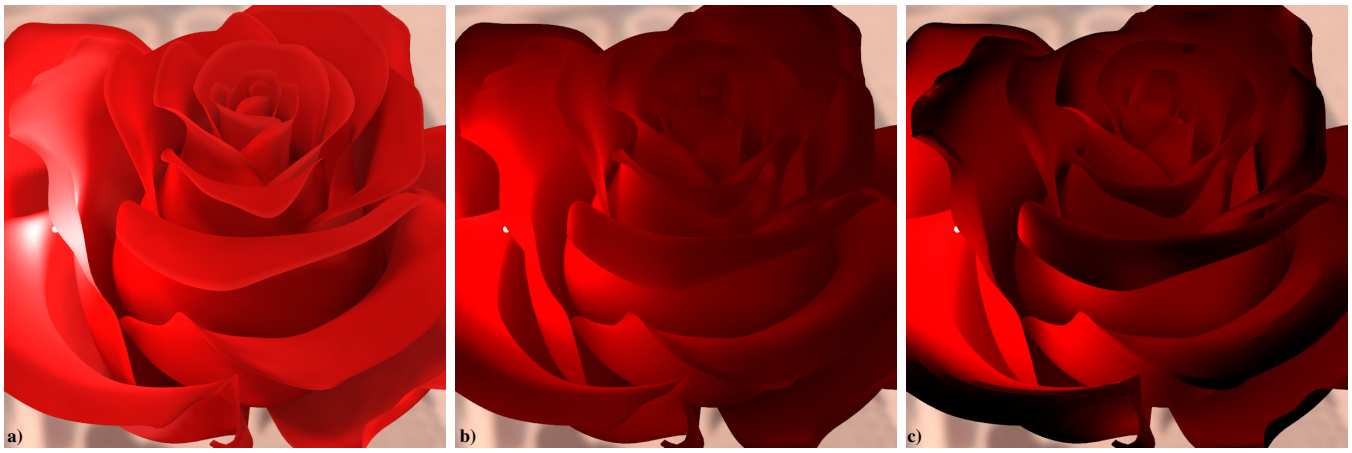


**Figure 9:** *a) A rose model (72k polygons) rendered with the lighting from St. Peter's Basilica environment map (debevec.org) and a point light source between the petals. b) The missing light transport component if only subsurface scattering is simulated. c) The global illumination if the rose was opaque and only diffuse inter-reflections (classical radiosity) is simulated.*

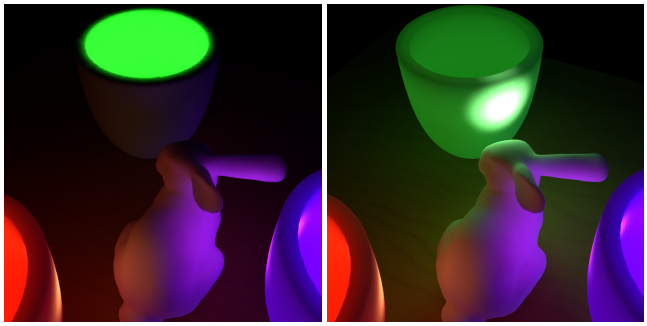

**Figure 11:** *Renderings of the "bowls and bunny" scene where one bowl is changed from being opaque (left) to being very translucent (right). Notice the color bleeding differences.*

inter-reflection plays its role, this may mean either $(\mathbf{I}-\mathbf{F})^{-1}\mathbf{SL}$ or $\mathbf{SL}+(\mathbf{I}-\mathbf{F})^{-1}\mathbf{L}$ with our notation. The first heuristic method only models one bounce of translucency inter-reflection and later bounces are substituted with diffuse inter-reflection, while the second one completely considers diffuse inter-reflection and composites the final rendering by adding scattering and diffuse inter-reflection. Both methods cause noticeable errors for multiple layers of translucent objects or objects with different diffuse and scattering colors. Figure 12 shows comparisons between our method and these two heuristics. The scene is illuminated by three spot-lights, one inside each bowl, and we set the diffuse colors to match the colors of the scattering profiles. Only our method can accurately capture all the light transport within the bowls and between the bowls and the bunny.

**Performance:** For simultaneous relighting and translucency change, we use an iterative sparse linear system solver (typically 5-10 iterations) in every frame. We implement a GPU numerical solver with CUDA SPARSE library. Our CUDA solver achieves a reasonable performance boost, with 5-10 fps for a scene with $40k$ polygons, while the CPU implementation runs at 1-2 fps. For the same scene, the precomputation of form-factor and subsurface scattering matrices takes about 10 minutes. Since our algorithm is a object-space method, changing viewpoint is real-time for free. If only relighting is considered, precomputing matrix $\mathbf{T}=(\mathbf{I}-\mathbf{SF})^{-1}\mathbf{S}$ can improve the relighting speed to 50-80 fps, achieving real-time rates. Comparing the performance with two related works in interactive rendering of translucent objects, our relighting speed is faster than [Wang et al. 2005], and comparable to that reported in Wang et al.'s work [Wang et al. 2008b]. However, our method models the light interactions between translucent objects and the environment while the two methods focus on rendering individual translucent objects. In addition, the precomputation time of our method is significantly smaller than that of these two methods (minutes vs. hours).

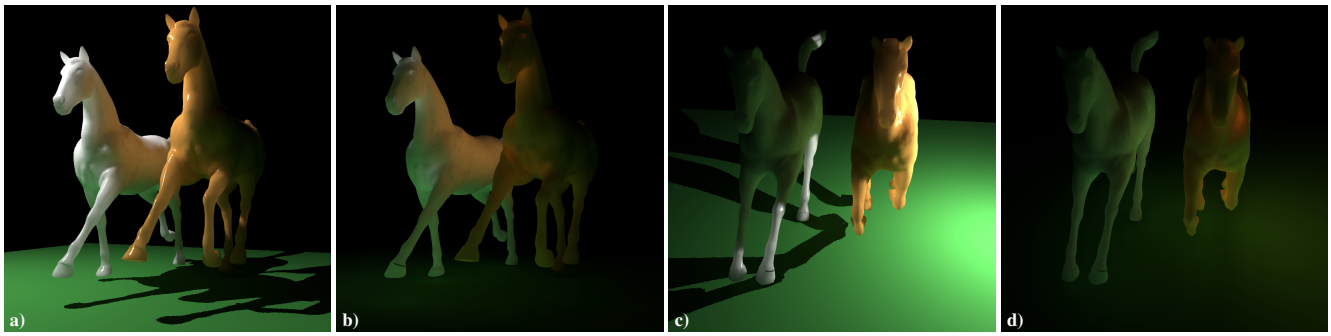


**Figure 10:** *(a) and (c): Two renderings of the "two horses" animation. Notice the green and brown colors on the white horses due to inter-reflection. (b) and (d): The missing light transport components if only subsurface scattering is simulated.*

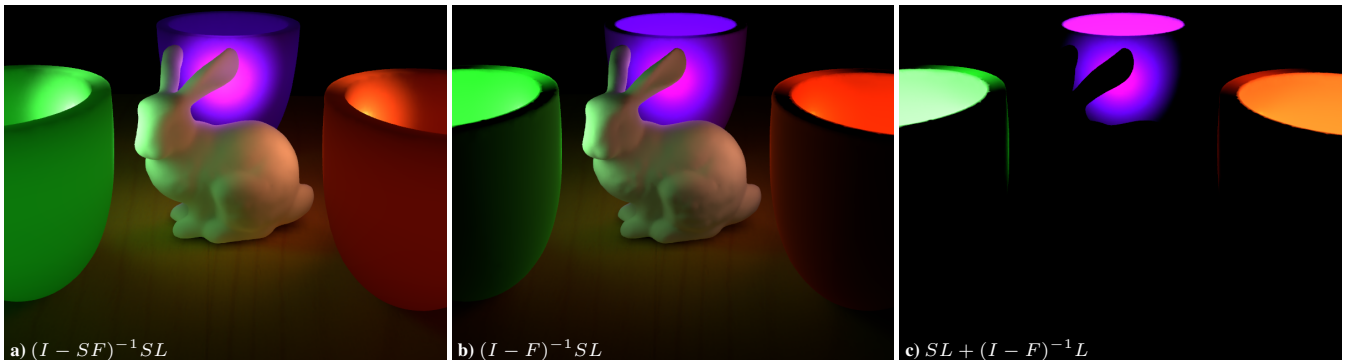


**Figure 12:** *Comparisons between our model and two heuristic methods. a) Our model can correctly capture the intertwined light transport of inter-reflections and subsurface scattering. b) The first heuristic method can only capture one bounce of translucent inter-reflection, and therefore the outside of the bowls and the bunny are much darker than the correct solution. The geometry detail on the bunny is more obvious than our method. c) The second heuristic method uses diffuse inter-reflection to replace translucent inter-reflections and therefore light cannot scatter outside the bowls at all.*

| Scene | # of polys | Precomputation | Memory | Relighting only (*fps*) | Relighting / Change-in-Translucency (*fps*) |
|---|---|---|---|---|---|
| Cornell box | 24640 | 4 min+3 min | 800MB | 90 | 14 / 4 |
| Bowls and bunny | 73256 | 10 min+12 min | 2.7GB | 28 | 3 / 0.25 |
| Rose | 72474 | 15 min+20 min | 2.7GB | 33 | 3 / 0.5 |
| Two horses | ~ 63000 | 9 min + 10 min | 2.4GB | 43 | 5 / 0.4 |

**Table 1:** *Performance summary of the four test scenes. The precomputation time includes both form-factor and subsurface scattering matrices. Column 5 lists the relighting frame-rates if computing the inverse, while the last column includes the frame-rates using our linear solver. Note that the values of the "two horses" scene are listed for each frame, not the whole animation. As shown in previous work [Willmott et al. 1999], hierarchical links can further reduce the storage to less than 200MB.*

**Storage:** The downside of any radiosity-based method is the memory consumption. Fortunately, due to occlusions in the geometry and locality of subsurface scattering, both $\mathbf{F}$ and $\mathbf{S}$ are sparse. By further thresholding small fractional numbers, we are able to fit the memory into the graphics card without introducing visible difference. Previous work of face cluster radiosity [Willmott et al. 1999] shows that it is able to consume only 120MB memory for a 2.7 million polygonal mesh by adopting volume clustering, therefore similar hierarchical links can further reduce our storage.

## 6 Conclusions and Limitations

We presented an analytic model to combine two specific forms of light transport — diffuse inter-reflections and isotropic subsurface scattering. Our approach is simple, easy to implement, extends classical radiosity with little additional cost and can capture interesting effects due to homogeneous and heterogeneous translucency, relighting translucent scenes and controlling object translucency, at near interactive rates. Further speed-ups can be achieved by using hierarchical and adaptive implementations [Hanrahan et al. 1991; Willmott et al. 1999], which we will leave as future work. That said, our approach does share the shortcomings of classical radiosity techniques as compared to ray-tracing methods. Hence, multi-bounces of specular reflection, caustics and volumetric scattering cannot be modeled and large meshes will require prohibitively high storage for today's GPUs and PCs. In the future, we will explore extending the model to interactively deform and animate translucent objects, without significant precomputation.

Finally, we requested an expert artist to use a commercial rendering tool (3D Studio Max with rendering engine V-Ray) to create the rendering of the bunny and bowls scene. The expert artist took approximately 2 hours to generate a single rendering qualitatively similar to our model. While a real user study is required, this sample experiment demonstrates the practicality of our work.

## Acknowledgments

The authors wish to thank the anonymous reviewers for their helpful comments. We also thank Shuo Wang and Changyu Diao for creating some of the models. Yu Sheng and Srinivasa G. Narasimhan were funded in parts by NSF CAREER award IIS-0643628, NSF grant IIS-0964562, and ONR grant N00014-11-1-0295. Yulong Shi and Lili Wang were supported in parts by NSFC grants 61272349 and 61190121.

## References

ARBREE, A., WALTER, B., AND BALA, K. 2008. Single-pass scalable subsurface rendering with lightcuts. *Computer Graphics Forum 27*, 2, 507–516.

ARBREE, A., WALTER, B., AND BALA, K. 2011. Heterogeneous subsurface scattering using the finite element method. *IEEE Trans on Vis. and Computer Graphics 17*, 956–969.

CARR, N. A., HALL, J. D., AND HART, J. C. 2003. GPU algorithms for radiosity and subsurface scattering. In *Graphics Hardware 2003*, 51–59.

CHRISTENSEN, P. H., LISCHINSKI, D., STOLLNITZ, E. J., AND SALESIN, D. H. 1997. Clustering for glossy global illumination. *ACM Transactions on Graphics 16*, 1 (Jan.), 3–33.

COHEN, M., AND WALLACE, J. 1993. *Radiosity and realistic image synthesis*. Academic Press Professional, Inc., USA.

COHEN, M., CHEN, S. E., WALLACE, J., AND GREENBERG, D. 1988. A progressive refinement approach to fast radiosity image generation. In *Proceedings of SIGGRAPH*, 75–84.

DIJKSTRA, E. W. 1959. A note on two problems in connexion with graphs. *Numerische Mathematik 1*, 269–271.

DONNER, C., AND JENSEN, H. W. 2005. Light diffusion in multi-layered translucent materials. *ACM Transactions on Graphics 24*, 3 (Aug.), 1032–1039.

DONNER, C., AND JENSEN, H. W. 2007. Rendering translucent materials using photon diffusion. In *Rendering Techniques 2007: 18th Eurographics Workshop on Rendering*, 243–252.

GOESELE, M., LENSCH, H. P. A., LANG, J., FUCHS, C., AND SEIDEL, H.-P. 2004. Disco: acquisition of translucent objects. *ACM Transactions on Graphics 23*, 3 (Aug.), 835–844.

GORAL, C., TORRANCE, K., GREENBERG, D., AND BATTAILE, B. 1984. Modelling the interaction of light between diffuse surfaces. In *Proc. of SIGGRAPH*, 213–222.

HANRAHAN, P., SALZMAN, D., AND AUPPERLE, L. 1991. A rapid hierarchical radiosity algorithm. In *Proc of SIGGRAPH*.

JAKOB, W., ARBREE, A., MOON, J. T., BALA, K., AND MARSCHNER, S. 2010. A radiative transfer framework for rendering materials with anisotropic structure. *ACM Transactions on Graphics 29*, 4 (July), 53:1–53:13.

JENSEN, H. W., AND BUHLER, J. 2002. A rapid hierarchical rendering technique for translucent materials. *ACM Transactions on Graphics 21*, 3 (July), 576–581.

JENSEN, H. W., LEGAKIS, J., AND DORSEY, J. 1999. Rendering of wet materials. In *Eurographics Rendering Workshop*.

JENSEN, H. W., MARSCHNER, S. R., LEVOY, M., AND HANRAHAN, P. 2001. A practical model for subsurface light transport. In *Proceedings of ACM SIGGRAPH*, 511–518.

LENSCH, H. P. A., GOESELE, M., BEKAERT, P., KAUTZ, J., MAGNOR, M. A., LANG, J., AND SEIDEL, H.-P. 2002. Interactive rendering of translucent objects. In *10th Pacific Conf. on Computer Graphics and Applications*, 214–224.

MERTENS, T., KAUTZ, J., BEKAERT, P., SEIDEL, H.-P., AND REETH, F. V. 2003. Interactive rendering of translucent deformable objects. In *Eurographics Symposium on Rendering: 14th Eurographics Workshop on Rendering*, 130–140.

MITCHELL, J. S. B., MOUNT, D. M., AND PAPADIMITRIOU, C. H. 1987. The discrete geodesic problem. *SIAM J. Comput. 16* (August), 647–668.

NG, R., RAMAMOORTHI, R., AND HANRAHAN, P. 2003. All-frequency shadows using non-linear wavelet lighting approximation. *ACM Transactions on Graphics 22*, 3 (July), 376–381.

NICODEMUS, F. E., RICHMOND, J. C., HSIA, J. J., GINSBERG, I. W., AND LIMPERIS, T. 1977. Geometrical considerations and nomenclature for reflectance. *Science And Technology 160*.

PHARR, M., AND HANRAHAN, P. M. 2000. Monte carlo evaluation of non-linear scattering equations for subsurface reflection. In *Proceedings of ACM SIGGRAPH 00*, 75–84.

PREMOZE, S., ASHIKHMIN, M., TESSENDORF, J., RAMAMOORTHI, R., AND NAYAR, S. 2004. Practical Rendering of Multiple Scattering Effects in Participating Media. In *Eurographics Symposium on Rendering*.

RUSHMEIER, H., AND TORRANCE, K. 1987. The zonal method for calculating light intensities in the presence of a participating medium. In *Computer Graphics (Proceedings of SIGGRAPH 87)*, 293–302.

RUSHMEIER, H. E., AND TORRANCE, K. E. 1990. Extending the radiosity method to include specularly reflecting and translucent materials. *ACM Transactions on Graphics 9*, 1 (Jan.), 1–27.

SLOAN, P.-P., KAUTZ, J., AND SNYDER, J. 2002. Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. *ACM Transactions on Graphics 21*, 3 (July), 527–536.

SLOAN, P.-P., HALL, J., HART, J., AND SNYDER, J. 2003. Clustered principal components for precomputed radiance transfer. *ACM Transactions on Graphics 22*, 3 (July), 382–391.

WALLACE, J., COHEN, M., AND GREENBERG, D. 1987. A two-pass solution to the rendering equation: A synthesis of ray tracing and radiosity methods. In *Proc. of SIGGRAPH*, vol. 21(4), 311–320.

WANG, R., TRAN, J., AND LUEBKE, D. 2005. All-frequency interactive relighting of translucent objects with single and multiple scattering. *ACM Transactions on Graphics 24*, 3 (Aug.), 1202–1207.

WANG, J., ZHAO, S., TONG, X., LIN, S., LIN, Z., DONG, Y., GUO, B., AND SHUM, H.-Y. 2008. Modeling and rendering of heterogeneous translucent materials using the diffusion equation. *ACM Transactions on Graphics 27*, 1 (Mar.), 9:1–9:18.

WANG, R., CHESLACK-POSTAVA, E., WANG, R., LUEBKE, D., CHEN, Q., HUA, W., PENG, Q., AND BAO, H. 2008. Real-time editing and relighting of homogeneous translucent materials. *The Visual Computer 24*, 7 (July), 565–575.

WILLMOTT, A., HECKBERT, P., AND GARLAND, M. 1999. Face cluster radiosity. In *Eurographics Rendering Workshop 1999*.