Screen-space subsurface scattering rendering

Jose I. Echevarria[1]
joseignacioechevarria@gmail.com

Adolfo Munoz[1]
adolfo@unizar.es

Francisco J. Seron[1,2]
seron@unizar.es

Diego Gutierrez[1,2]
diegog@unizar.es

[1]Universidad de Zaragoza

[2]Instituto de Investigacion
en Ingenieria de Aragon

Abstract

Existing subsurface scattering rendering algorithms try to reproduce the phenomenon aiming at physically accurate results. Recent research has demonstrated that in some scenarios, to make some assumptions about the way we perceive translucency helps us speeding up calculus achieving real-time results. In this paper we present a new algorithm based on previous perceptual studies general enough to work with measured and user-defined data. The results obtained make the algorithm interesting in scenarios were a plausible appearance of translucency is enough.

1 Introduction

Translucency is a challenging phenomenon to simulate due to its complex subsurface light transport. In the last decade, breakthroughs were presented [JMLH01], making the problem more affordable and enabling its simulation in efficient ways. Due to the results obtained with offline rendering techniques, recent research in the area has focused on trying to reproduce them in real-time. In this case, we can find two main groups: approaches that try to find representations and data structures which help to speed up rendering times, and approaches that try to simulate subsurface scattering in a more perceptual way. In this paper we are going to focus in the last group.

We leverage the fact that the human visual system is not good calculating inverse optics, instead of this we perceive translucency based on image heuristics and visual cues [FB05]. This enable us to calculate subsurface scattering more efficiently by cheating our brain with physically plausible renders. Inspired by the work of Jimenez et al. [JSG09, JWSG09], we present a screen-space rendering technique looking for a more general solution, allowing the use of general diffusion profiles and enabling the simulation of a wide range of materials apart from skin.

Our approach is based on irradiance convolutions over a multi-layered representation of the object, which is general enough to obtain plausible depictions of translucent objects based on the diffusion approximation. Our goal is not a real-time algorithm, instead we are exploring the extendability of current real-time techniques to more general scenarios. Thus, our algorithm is implemented in CPU leaving the door opened for future improvements and GPU implementations, if possible.

The results obtained show that this technique is general enough to work with both measured data from previously publications [JMLH01, NGD*06] and user-defined materials.

2 Previous work

Rendering: The dipole method is one of the most popular techniques for simulating the appearance of subsurface scattering for translucent materials [JMLH01]. This approach formulates a BSSRDF by combining an exact solution for single scattering with a dipole point source diffusion approximation to simulate multiple scattering. We use this and subsequent work extending the original model [DJ05], as the basis of our translucent material model. A number of real-time algorithms to simulate the appearance of translucency exploit consumer level graphics hardware, however restrict the types of diffusion profiles possible [dLE07, CLH*08, JSG09, JWSG09]. We aim to efficiently simulate more general diffusion profiles compared to these earlier methods.

Visual Perception: A parallel line of research into how humans visually perceive shape and material properties shows that while humans have a good intuition for natural lighting, they cannot establish the exact correspondence between shape, reflectance and patterns of lighting [BKY99, FDA03, OCS05]. A detailed study of the cues that affect human perception of translucency concludes that objects composed of such material appear more realistic if they present specular highlights [FB05]. Further, specularity aids shape perception of translucent materials, which would otherwise lose visual detail due to the softening effects of sub-surface scattering.

3 Subsurface scattering rendering

Our algorithm is based on the diffusion approximation [JMLH01] which defines multiple subsurface scattering as:

$$L_m(\vec{x}_{out}, \vec{\omega}_{out}) = \frac{1}{\pi} F_t(\frac{n_{ob}}{n_{med}}, \vec{\omega}_{out}) \cdot \int_A R_d(\|\vec{x}_{out} - \vec{x}_{in}\|) I(\vec{x}_{in}) dA(\vec{x}_{in}) \quad (1)$$

where L_m is the exitant radiance, F_t is the Fresnel transmission term, A defines the surface area of the object, $\vec{x}_{in}$ and $\vec{x}_{out}$ define the incident and exitant point of light respectively, ω_{in} and ω_{out} define the incident and outgoing light directions and n_{ob} and n_{med} define the indices of refraction of the object and the medium. R_d is a one-dimensional function called *diffusion profile* that defines the properties of the material regarding subsurface scattering. Several models for this function can be found in different works, such as the dipole model [JMLH01], which will be chosen in order to use their captured materials. I defines the irradiance, that can be computed as follows:

$$\mathrm{I}(\vec{x}_{in}) = \int_{\Omega} F_t(\tfrac{n_{ob}}{n_{med}}, \vec{\omega}_{in}) L(\vec{x}_{in}, \vec{\omega}_{in})(\vec{n}_{in} \cdot \vec{\omega}_{in}) d\vec{\omega}_{in} (2)$$

where Ω refers to the whole hemisphere of incident light, L represents incident radiance and $\vec{n}_{in}$ is the normal of the surface at $\vec{x}_{in}$. This is computed for all color channels (RGB).

We assume that the object is *optically thick*, an assumption that has been done before [XGLJH07, JSG09]. Single scattering in optically thick materials is negligible compared to multiple scattering, and therefore Equation 1 models all subsurface scattering for such materials. As it is shown further in the text, this simple assumption enables us to simulate light transport by means of image convolutions.

We first define a set of equally spaced parallel planes that cut the object along its volume. Although our algorithm works for a generic orientation of those planes, the best results are achieved when these are parallel to the projection plane of the camera. From now on, we consider that the orientation of these planes is the optimal one and therefore we work on screen space.

All the irradiance that is incident along the surface of the object is stored at the closest layers, and then light transport is calculated by considering the diffusion profile and performing convolutions between layers. Then the final result integrates the contributions from all layers. This algorithm is sufficiently general that it can also be used with measured scattering data [JMLH01, NGD*06], and several user-defined diffusion profiles by setting up absorption and extinction coefficients and using the dipole model

3.1 Algorithm

For the definition of this algorithm, the coordinate system is the one of the camera, where the x and y axis are horizontal and vertical axis, and z axis represents the direction perpendicular to the projection plane. We subdivide the scene by defining a series of n_l layers equally spaced along the z axis of the shape of the object, and parallel to the image plane. Each layer is then located at a distance z^l given by:

$$z^l = \min(z) + l\Delta z \qquad (3)$$

where $\Delta z = (\max(z) - \min(z))/(n_l - 1)$, $l = 0..n_l - 1$ and $\min(z)$ and $\max(z)$ represent the limits of the bounding box of the object in the z axis. Visually pleasing results are achieved with n_l varying between 4 and 8, more layers increase accuracy but differences are not perceived. Our diffusion approximation is similar in spirit to the work of Donner and Jensen [DJ05]. Assuming that all interactions between pairs of layers are due to multiple scattering, we rely on convolution to compute the contribution between them. Note that in Donner and Jensen's approach they attempt to simulate multilayered materials by assuming that they are differentially parallel, although the surface of the corresponding object might not be plane.

At each of this layers, we define its corresponding *incident irradiance map* I^l_{in}. Working in screen space, our results show that a resolution between 4 and 8 times smaller than the resolution of the resulting image is a good compromise between final image quality and computation times, depending on the properties of the material (the diffusion profile) and on the geometry of the object. We first choose points on the surface as Jensen et al. [JB02]. Next, we calculate irradiance at those points (methods will depend on the kind of source lights in the scene). Finally, for every layer, we distribute the irradiance along the corresponding incident irradiance maps (see Figure 1, left):

$$I^l_{in}(x,y) = \sum_{x_{in}} I(\vec{x}_{in}) \max\left(1 - \frac{|z^l - z_{in}|}{\Delta z}, 0\right) \qquad (4)$$

where $I(\vec{x}_{in})$ is obtained from Equation 2.

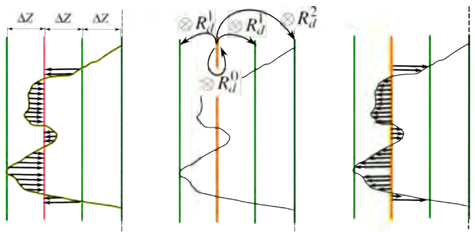

Figure 1: From left to right: Projecting the incoming irradiance onto one layer; Irradiance transfer between layers by convolutions; Projecting the outgoing irradiance of a layer back onto the object.

We generate n_l convolution maps R^e_d, each one representing the effect of the incoming irradiance at a specific layer, on the outgoing radiance at a layer at distance $e = m\Delta Z$ with $m = 0, ...n_l - 1$. These convolution maps are generated from the function R_d:

$$\mathrm{R}^e_d(x,y) = R_d\left(\sqrt{\left(k_x\left(x - \tfrac{w}{2}\right)\right)^2 + \left(k_y\left(y - \tfrac{h}{2}\right)\right)^2 + (e\Delta z)^2}\right) \qquad (5)$$

where w and h represent the width and height of the down-sampled incoming irradiance maps. k_x and k_y are scale factors that relate the size of the irradiance maps with the size of the geometry of the object. The outgoing irradiance map at each layer (I^l_{out}) is then obtained by convolving it with each R^e_d (see Figure 1, middle), yielding:

$$I^l_{out}(x,y) = \sum_{\forall i, e=|i-l|} I^i_{in}(x,y) \otimes R^e_d(x,y) \qquad (6)$$

For efficiency reasons, this convolution is computed using the discrete fast Fourier method implemented in the FFTW library [FJ05]. In Fourier space convolutions become per-pixel complex multiplications, which are by far much more efficient than computing the whole convolution.

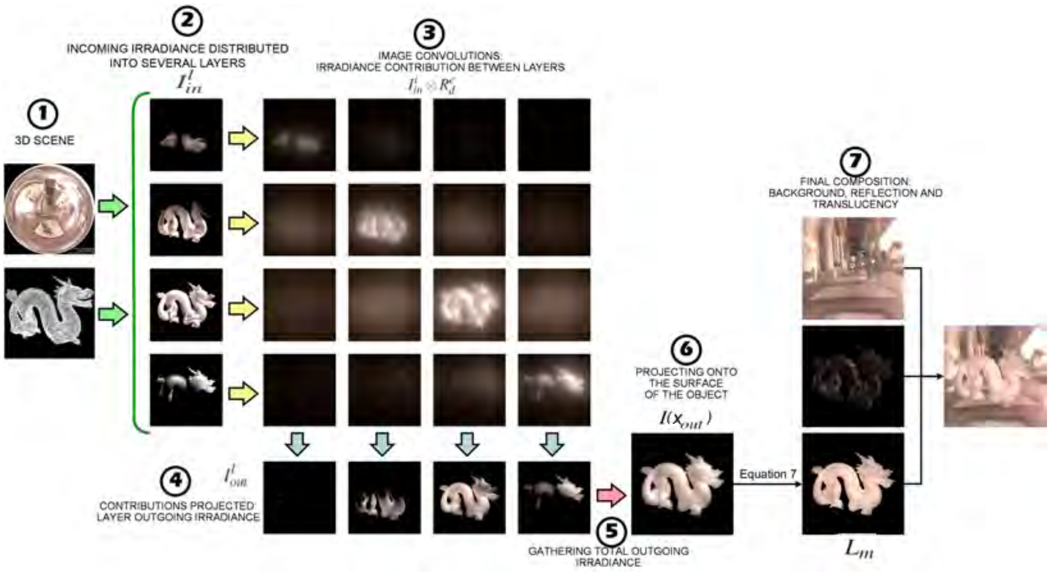
2
INCOMING IRRADIANCE DISTRIBUTED INTO SEVERAL LAYERS
I^l_{in}
3
IMAGE CONVOLUTIONS: IRRADIANCE CONTRIBUTION BETWEEN LAYERS
$I^l_{in} \otimes R^c_d$
1
3D SCENE
7
FINAL COMPOSITION: BACKGROUND, REFLECTION AND TRANSLUCENCY
6
PROJECTING ONTO THE SURFACE OF THE OBJECT
$I(x_{out})$
4
I^l_{out}
CONTRIBUTIONS PROJECTED LAYER OUTGOING IRRADIANCE
Equation 7
5
GATHERING TOTAL OUTGOING IRRADIANCE
L_m

Figure 2: Overview of the rendering algorithm

The final outgoing radiance at any point x_{out} is computed by using Equation 1 as follows:

$$L_m(\vec{x}_{out}, \omega_{out}) = \frac{1}{\pi} F_t(\frac{n_{ob}}{n_{med}}, \omega_{out}) I(x_{out}) \tag{7}$$

where the exitant irradiance $I(x_{out})$ is computed from trilinear interpolation, considering the four closest pixels of the two outgoing irradiance maps from the two closest layers (see Figure 1, right).

Specular highlights are included afterwards by simply adding the corresponding specular Phong addend in order to enhance the perception of translucency [FB05]. We can see the complete pipeline in Figure 2.

4 Results and discussion

The technique presented in this paper is versatile enough to model a wide variety of materials. Figure 4 shows buddhas made of ketchup and low fat milk, renders are illuminated with *Uffizi Gallery* environment [YDMH99].

Rendering takes around 6 seconds for a 1024x1024 input image, with irradiance layer maps at 256x256 and with four layers involved. We have applied the photographic tone-mapping operator [RSSF02] to all the images for display purposes.

By changing RGB absorption and extinction coefficients in the dipole function, or entirely replacing the R_d model, we can create completely new material appearance. Figure 5 show a user-defined jade-like buddha illuminated from behind with a directional light. Another example can be found in the same figure, in this case a user-defined marble-like dragon rendered in *Galileo's Tomb* environment [YDMH99].

Finally, we show another example of application in Figure 3. In this case, our rendering algorithm has been applied over estimated geometry yielding better results than [KRFB06] and similar to other renders with real 3D geometry.

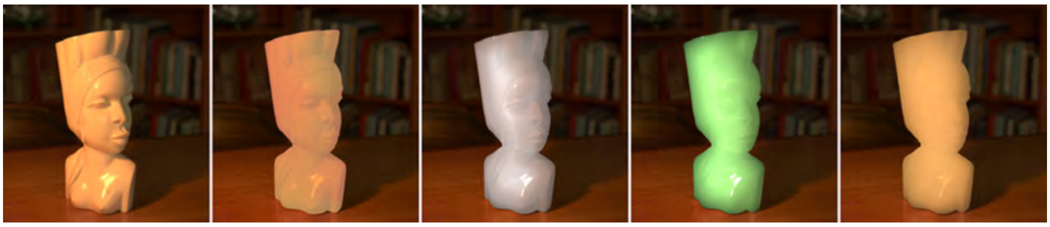

Figure 3: Demonstration of our method in an image-based editing context. From left to right: original image and translucency simulation from [KRFB06] (images and geometry courtesy of the authors), marble, jade and wax simulation with our method. Note that [KRFB06] cannot simulate any specific material, while we use both measured data (marble, from [JMLH01]) and new user-defined materials.

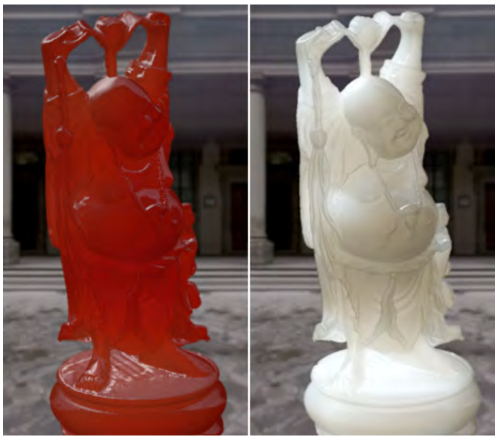

Figure 4: Two renders of buddhas made of measured ketchup [JMLH01] and low fat milk [NGD*06].

5 Limitations and future work

Current limitations of our method are relative to rendering times. Due to our CPU implementation, convolutions take a big part of the rendering time, so we use a lower resolution for the irradiance layer maps in order to accelerate this step. This down-sampling affects the final appearance of the render and this could be a problem with fast decaying diffusion profiles, where the softening effects of our approach could yield a visible loss of detail. Using layers with variable resolution and distribution depending on the geometry could be studied.

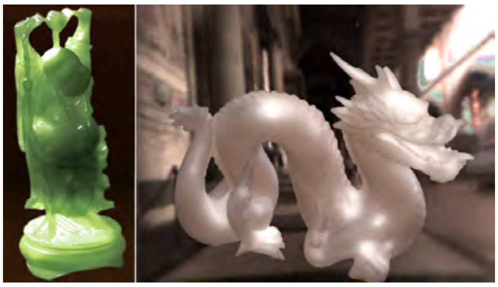

Figure 5: Two additional results, showing user-defined jade and marble simulations.

Recent FFT GPU implementations [GBDSM08] could be used to speed up calculus, this could reduce greatly rendering times, and while real-time could still be a challenge, performance gains would give us more interactivity.

Also, our tests are restricted to the dipole model and optically-thick materials. It would be interesting the simulation of optically thin materials.

Finally, we have seen in Figure 3 that our algorithm can work with estimated geometry. It would be interesting to study the applicability of our rendering method with existing shape estimation techniques, as this could extend the current repertoire of image-based editing tools.

6 Conclusions

We have presented a new rendering algorithm for translucency that works on screen-space and is general enough to work with a wide range of measured and user-defined materials. Despite the approximations and assumptions made in our simulations, they succeed on reproducing a satisfying appearance of translucency. We think that our work could inspire future real-time algorithms or image-based editing tools.

7 Acknowledgements

This research has been partially funded by the Spanish Ministry of Science and Technology (TIN2007-63025) and the Gobierno de Aragon (projects OTRI 2009/0411 and CTPP05/09). Jose I. Echevarria was funded by a research grant from the Instituto de Investigacion en Ingenieria de Aragon.

References

[BKY99] Belhumeur P. N., Kriegman D. J., Yuille A. L.: The bas-relief ambiguity. *International Journal of Computer Vision 35*, 1 (1999), 33–44.

[CLH*08] Chang C.-W., Lin W.-C., Ho T.-C., Huang T.-S., Chuang J.-H.: Real-time translucent rendering using gpu-based texture space importance sampling. *Computer Graphics Forum 27*, 2 (2008), 517–526.

[DJ05] Donner C., Jensen H. W.: Light diffusion in multi-layered translucent materials. In *SIGGRAPH* (Los Angeles, California, 2005), ACM, pp. 1032–1039.

[dLE07] d'Eon E., Luebke D., Enderton E.: Efficient rendering of human skin. In *Eurographics Symposium on Rendering* (Grenoble, France, 2007), Kautz J., Pattanaik S., (Eds.), Eurographics Association, pp. 147–157.

[FB05] Fleming R. W., Bülthoff H. H.: Low-level image cues in the perception of translucent materials. *ACM Transactions on Applied Perception (TAP) 2*, 3 (2005), 346–382.

[FDA03] Fleming R. W., Dror R. O., Adelson E. H.: Real-world illumination and the perception of surface reflectance properties. *Journal of Vision 3*, 5 (2003), 347–368.

[FJ05] Frigo M., Johnson S. G.: The design and implementation of FFTW3. *Proceedings of the IEEE Special Issue on Program Generation, Optimization, and Platform Adaptation 93*, 2 (2005), 216–231.

[GBDSM08] Govindaraju N. K., Brandon L., Dotsenko Y., Smith B., Manferdelli J.: High performance discrete Fourier transforms on graphics processors. In *SC '08: Proceedings of the 2008 ACM/IEEE conference on Supercomputing* (Austin, Texas, 2008), IEEE Press, pp. 1–12.

[JMLH01] Jensen H. W., Marschner S. R., Levoy M., Hanrahan P.: A practical model for subsurface light transport. In *SIGGRAPH* (Los Angeles, California, 2001), ACM, pp. 511–518.

[JB02] Jensen H. W., Buhler J.: A rapid hierarchical rendering technique for translucent materials. In *SIGGRAPH* (San Antonio, Texas, 2002), ACM, pp. 576–581.

[JSG09] Jimenez J., Sundstedt V., Gutierrez D.: Screen-Space Perceptual Rendering of Human Skin. *ACM Trans. Appl. Percept. 6*, 4 (2009), ACM, Article 23.

[JWSG09] Jimenez J., Whelan D., Sundstedt V., Gutierrez D.: Real-Time Realistic Skin Translucency. To appear in *Journal of IEEE Computer Graphics and Applications.*, (2009).

[KRFB06] Khan E. A., Reinhard E., Fleming R. W., Bülthoff H. H.:

Image-based material editing. In *SIGGRAPH* (Boston, Massachusetts, 2006), ACM, pp. 654–663.

[NGD*06] Narasimhan S. G., Gupta M., Donner C., Ramamoorthi R., Nayar S. K., Jensen H. W.: Acquiring scattering properties of participating media by dilution. In *SIGGRPH* (Boston, Masschusetts, 2006), ACM, pp. 1003–1012.

[OCS05] Ostrovsky Y., Cavanagh P., Sinha P.: Perceiving illumination inconsistencies in scenes. *Perception 34*, 11 (2005), 1301–1314.

[RSSF02] Reinhard E., Stark M., Shirley P., Ferwerda J.: Photographic tone reproduction for digital images. In *SIGGRAPH* (San Antonio, Texas, 2002), ACM, pp. 267–276.

[XGLJH07] Xu K., Gao Y., Li Y., Ju T., Hu S-M.: Real-time Homogenous Translucent Material Editing. *Computer Graphics Forum 26*, 3 (2009), pp. 545–552.

[YDMH99] Yu Y., Debevec P., Malik J., Hawkins T.: Inverse Global Illumination: Recovering Reflectance Models of Real Ccenes from Photographs. In *SIGGRAPH* (Los Angeles, California, 1999), ACM, pp. 215–224.