Reconstruction of Illumination from Area Luminaires

Steven Collins[1]

ISG, Trinity College Dublin, Ireland

Abstract. This paper is concerned with the efficient reconstruction of illumination from area luminaires. We outline a 2-pass scheme; a light-pass, tracing ray bundles from the luminaires followed by a general eye-pass ray trace phase. Special attention is paid during the first pass to the reconstruction of shadows, both umbral and penumbral regions, for general luminaire geometries with no practical restriction on surface reflectance distributions. In order to minimise computation times, adaptive sampling techniques are used to concentrate effort in regions of high illumination gradients and in particular regions exhibiting discontinuities. Illumination functions are reconstructed from samples in a manner related to adaptive kernel density estimation techniques.

1 Introduction

Recently the illumination from specular surfaces has been separated from other transport mechanisms and dealt with as a pre-processing stage. Arvo [Arv86] traces rays from the light sources and records these rays on surfaces in *illumination maps*. Shirley [Shi90] combines this light-pass with a low resolution zonal radiosity pass and finally a distribution ray tracing eye-pass to compute a final solution. Both approaches suffer from being limited in practise to point light sources, due to noise problems associated with sampling area sources. Heckbert [Hec90] used bidirectional ray tracing and adaptive illumination maps to handle all transport paths at the expense of large *light quad-trees*. More recently, the work of Shirley et al. [Shi95] uses density estimation techniques (in a manner similar to that proposed here) and records illumination as particle hit points with the surfaces.

The method proposed in this paper borrows from previous work; pencil tracing [STN87] and Ray Casting for Radiosity Shadows [Ase92]. We extend our earlier efforts [Col94] which tracked the wavefront from a point light source to allow for wavefront tracking from area sources. Rays are sent from the area source (initially approximating the area source as a point source), and having determined a point of intersection, the intersected object acts as a *shadow/caustic lens*. Each ray has associated with it some $\Delta\Phi_s$, a fraction of the total source power. Having intersected an object, we determine the solid angle through which this energy has arrived and distribute reflected and transmitted rays distributed within the reflected/refracted solid angle. To compute shadows we cast backwards shadow rays which carry *negative power*, $-\Delta\Phi_s$, again distributed within the solid angle. In order to deposit the power on surfaces we use adaptive Gaussian kernels to

[1] The author may be contacted at `Steven.Collins@cs.tcd.ie` or on the web at `http://vangogh.cs.tcd.ie/scollins/scollins.html`

preserve detail in regions of high frequency illuminance gradients and minimise noise in areas of low gradient (effectively the adaptive kernel density estimation technique [Sil86]). We present an adaptive sampling scheme which is capable of sampling finely in area of sharp shadowing/caustics and coarsely in areas of diffuse reflection and umbral/penumbral shadow regions.

The illuminance stored in the illumination maps is used during a 2nd pass (the *eye pass*), in that rays incident on a surface will extract the radiance estimation from the illumination map, and then according to the BRDF of the surface, specularly reflected/transmitted rays will be fired. Note, however, that no shadow rays are required during this phase of the simulation.

2 Problem Definition

We first define the irradiance on a surface, $E(x)$, in terms of separate transport paths; direct illumination E_d and indirect illumination E_i:

$$E(x) = E_d(x) + E_i(x) = \int_{\Omega_S} L(\omega_s)\cos\theta d\omega_s + \int_{\Omega_{\bar{S}}} L(\omega_{\bar{s}})\cos\theta d\omega_{\bar{s}} \qquad (1)$$

Ω_S is the set of directions in which the light sources are directly visible and $\Omega_{\bar{S}}$ all other directions[2]. We can further classify E_d as the total irradiance from all light sources less the obstructed irradiance (where S' is the solid angle subtended by the sources, assuming unocclusion):

$$E_d(x) = \int_{\Omega'_S} L(\omega_s)\cos\theta d\omega_s - \int_{\Omega'_S} L(\omega_s)\overline{V(\omega_s)}\cos\theta d\omega_s \qquad (2)$$

The algorithm presented here will assume an un-occluded contribution to each dA from each light source and will then compensate by irradiating dA with negative power to compensate for the $V(\omega_s)$ term in Equation 2. In short, we cast rays from the luminaires, each ray carrying a fraction of the total source power Φ_s/n. Rays hitting surfaces contribute illuminance given by $E \equiv \frac{d\Phi}{dA}$.

Note that using the irradiance quantity in this way resolves the normal dichotomy normally associated with the treatment of point sources[3]. If the power-carrying rays hit a surface we shoot new rays distributed according to the BRDF of the surface to contribute indirect illuminance to other surfaces. However we also fire backwards shadow rays in the direction of the original ray, carrying power $\Phi_r = -\Phi_s/n$. These rays will approximate the occluded irradiance of Equation 2 on surfaces occluded by the initial surface.

The radiance at a point on a surface may now be expressed in terms of these quantities:

---

[2] This assumes that light will not be indirectly reflected from or transmitted through the light sources themselves, in general a reasonable approximation.

[3] For point sources of non-finite dA we must use *radiant intensity* which is related to power by $d\Phi \equiv I(\omega)d\omega$.

$$L(x,\omega') = L_e(x,\omega') + \int_{\Omega-\Omega_S} \rho(x,\omega,\omega')L(\omega)\cos\theta d\omega$$
$$+ \sum_{AllSources} \int_{\Omega_S} \rho(x,\omega_s,\omega')L_s(\omega_s)\cos\theta d\omega_s$$
$$- \sum_{AllSources} \int_{\Omega_S} \rho(x,\omega_s,\omega')L_s(\omega_s)\overline{V(x,\omega_s)}\cos\theta d\omega_s \quad (3)$$

3 Illumination Storage

To record the shadow and indirect illuminance from luminaires on surfaces we use *illumination maps* in the spirit of [Arv86], in favour of *radiosity textures* [Hec90]. Essentially we represent the illuminance on a surface as a projection onto some function basis B_i:

$$E(x) \approx \hat{E}(x) = \sum_{i=0}^{n-1} e_i B_i(x) \text{ where } e_i = \sum_{x=0}^{X} E(x)B_i(x) \quad (4)$$

This has the advantage that we can choose arbitrary bases. We currently employ a piecewise linear basis, but are investigating various wavelet bases to provide efficient storage of the illumination maps.

As rays deposit power in the illumination maps cells, the cell accumulates this power in its current total cached power: $\Phi_{old} = \Phi_{new} + \Phi_r$. During the eye-pass, the cached power in a cell of area A is converted to radiance using $L = \frac{\rho_r \Phi}{\pi A}$.

This is based on the assumption that future eye-pass rays will determine the specularly reflected/transmitted radiance and the illumination map stores only the energy that is diffusely reflected (thus $\rho_{BRDF}(\omega,\omega') = \frac{\rho_r}{\pi}$ where ρ_r is now a reflectance quantity).

3.1 Illumination Map Resolution

The resolution of the illumination map is critical, in that too coarse a resolution will filter all detail from the solution, and too fine a resolution will incur memory overheads, and introduce aliasing problems. We can guide the resolution determination either

(a) by using view dependent information and setting the resolution proportion to the projected area of the surface on the viewing plane [Hec90] or
(b) using a view independent scheme where resolution is adaptively chosen according to the illuminance gradients on the surface [Col94].

In our current implementation we adopt a view dependent approach, and choose a resolution $R_x = R_y = 2^n$ where n is chosen so that $A_{map} \approx \frac{A_{pix}}{2}$ where A_{map} is the area of an illumination map pixel projected into screen space and A_{pix} the area of a pixel in screen space.

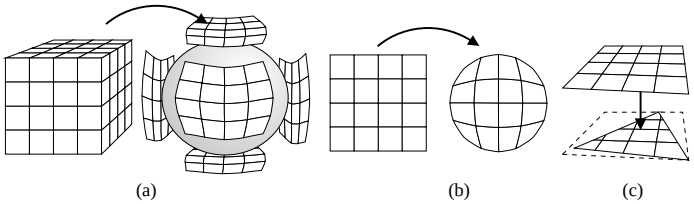

Fig. 1. Various approximately area-preserving parameter space mappings. (a) shows the cubic mapping onto a sphere, (b) an inverse radial stretch mapping a square to a disc and (c) a parallel projection for polygonal elements.

3.2 (u, v) Mapping

Illumination maps are defined with respect to parameter space (u, v) where $u \times v \subset \Re^2$. This mapping from parameter space to object space is not necessarily linear, and in fact rarely is. As a result, illumination map pixel area is not preserved and thus the radiance estimate will be skewed (for example near the poles in a spherical mapping). To counter this, primitive objects must have associated with them a mapping that is, or approximates, an area preserving mapping. To this end we use a cubic projection method for spheres, parallel projection for polygonal elements and *inverse radial stretch* of a square to a circle for discs (used as caps in solid cylinders). These mappings are shown in Figure 1. This highlights one of the main deficiencies of the approach. Ideally, we would like to store illuminance in an object independent fashion, perhaps volumetrically as in [War92]. This is currently under investigation.

4 Wavefront Tracking

Central to the algorithm is the tracking of rays from the luminaires into the environment. If the power associated with a ray is deposited as a single point in the illumination maps then, to minimise noise, we must shoot many rays or alternatively increase the size of the illumination map pixels. However, by associating each ray with a differential solid angle, $d\omega$ of the source, and through tracking the convergence/divergence of this $d\omega$ we can determine an *area of influence* for the ray when it hits a surface (ie: the area over which the power associated with the ray must be spread). Previous attempts to track the wavefront in such a way [Col94] required the use of expensive ray-tree caches, recording the intersection of each ray in the tree and correlating between neighbouring rays to determine a *spread factor*. As the ray tree increased in depth, the storage costs associated with this cache became prohibitive.

We propose a new approach to wavefront tracking which borrows much from *paraxial theory* [Hect87] of geometric optics and *pencil tracing* [STN87], a contemporary application of paraxial theory to computer graphics.

A *paraxial ray* is a ray propagating in the region of an *axial ray*. Each paraxial ray is represented by a *ray vector*, a four dimensional vector quantity $\psi = (x_1 x_2 \xi_1 \xi_2)^T$.

$(x_1 x_2)^T$ is the intersection of the paraxial ray and the plane $(\mathbf{u} \times \mathbf{v})$, where $(\mathbf{u}, \mathbf{v}, \mathbf{w})$ is an orthogonal coordinate system, with $\mathbf{w}$ co-incident with the direction of the axial ray. $(\xi_1 \xi_2)$ is the projection of the paraxial direction vector on

the ($\mathbf{u} \times \mathbf{v}$) plane (see Figure 2(a) for the associated geometry). We can now represent any optical interface as a 4×4 system matrix M (assuming coherent propagation media) if the deviation of the paraxial ray from the axial ray is small enough. Thus the transformed paraxial ray becomes $\psi' = M\psi$.

To derive the system matrix M^r, [STN87] approximate the surface as a paraboloid (similar to the tangent plane approximation of [Bar86]). A maximum paraxial ray spread angle is then approximated using a tolerance based on pixel width and ray sampling interval. Herein lies a difficulty with the approach. The convergence/divergence of the wavefront is a feature that we wish to adapt to and indeed exploit to reduce the number of rays required. Wavefront divergence (for example when the wavefront is incident on a convex mirror) should result in a reduced density of rays tracked, whereas focused light should result in higher sampling densities to accurately capture the illumination detail.

Rather than approximate optical interfaces as transfer matrices, we explicitly build a paraxial ray description of the differential wavefront $\Delta\omega$ being represented by the axial ray. We define a *slice* as $\psi \triangleq \{(x\boldsymbol{\xi}), (x_u\boldsymbol{\xi}_u), (x_v\boldsymbol{\xi}_v)\}$.

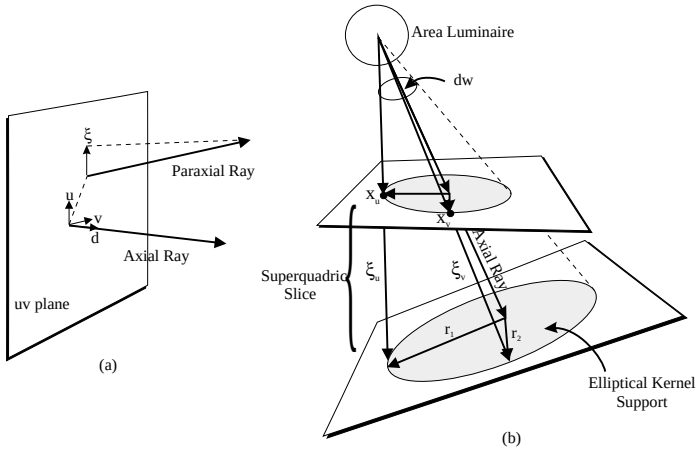

Fig. 2. (a) The geometry of pencil tracing. (b) The superquadric slice.

The 3 rays $(x\boldsymbol{\xi})$, $(x_u\boldsymbol{\xi}_u)$ and $(x_v\boldsymbol{\xi}_v)$ define a superquadric slice (see Figure 2(b)). By tracking the rays through the environment we track the superquadric as it experiences deformation at media interfaces. When the slice hits a surface, the slice's power is deposited over the area defined by the intersection of the slice and the surface. In order to simplify the determination of this region, we assume planarity at the interface giving an super-conical region of influence.

Effectively we are using the approach of the variable kernel method for the reconstruction of a density function [Sil86]:

$$f(t) \approx \hat{f}(t) = \frac{1}{n}\sum_{j=1}^{n} \frac{1}{hd_{j,k}} K\left(\frac{t - X_j}{hd_{j,k}}\right) \tag{5}$$

K is some kernel function such that $\int_{-\infty}^{\infty} K(x)dx = 1$, h a smoothness parameter and $d_{j,k}$ a space variant density estimator. Usually this method requires an initial density estimation to determine $d_{j,k}$. However, in determining the region of influence of the slice we have determined an effective measure of $d_{j,k}$.

For the kernel shape we use an elliptical Gaussian. Using the method proposed in [Got93] the elliptical Gaussian *splat* of energy can be computed from an appropriate set of basis kernels:

$$K(r_1, r_2, \theta)(x, y) = \sum_{i,j,k} c_{ijk}(r_1, r_2, \theta) B_{ijk}(x, y) \tag{6}$$

$K(r_1, r_2, \theta)$ represents a canonical elliptical kernel of radii r_1, r_2, rotated by θ. The coefficients, c_{ijk} are computed using singular value decomposition techniques.

4.1 The Superquadric of Confusion

Given that all the superquadric slices have the centre of the luminaire as an apex, we can handle only point source effects. To extend the algorithm to allow area luminaires to be effectively simulated we are required to sample the surface area of the luminaire in addition to the solid angle through which the source radiates. This dramatically increases the noise level in the solution and requires significantly higher sampling rates. We now propose an extension to the superquadric slice tracing idea to cater for area luminaires.

We can observe that when a light-pass ray is cast from the luminaire with its origin at the centre of the luminaire, the point of intersection of this ray and a surface may be considered as a focal point for a bundle of rays. Behind the surface this bundle volume diffuses from the focal point. When incident on a receiving surface the bundle defines a *region of confusion*, a term borrowed from the notion of a *circle of confusion* when dealing with lens effects. The bundle of rays therefore define what we term the *superquadric of confusion*.

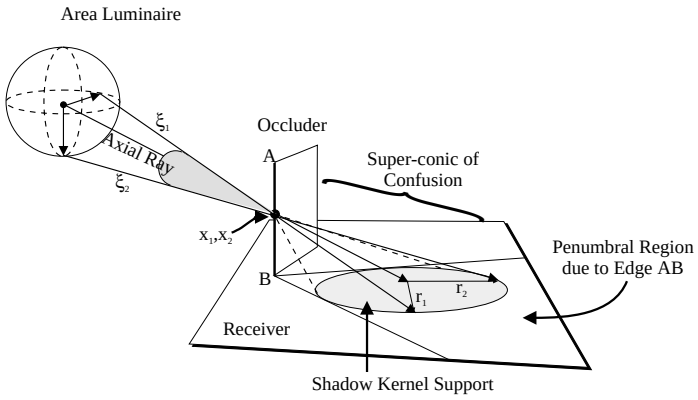

Fig. 3. (a) The superquadric of confusion resulting from a point on an occluder and a spherical light source.

This bundle of rays is no more than the rays defining the solid angle subtended by the source at the surface intersection point, as shown in Figure 3, for a spherical light source. This idea has been used in [Ase92] to partition the environment between receiver (the intersected surface) and the source according to the size of penumbral region that objects in the region will cause on the receiver, to determine how coarsely to sample those objects with a view to computing the shadows cast.

To use this we extend our definition of the slice to include 2 co-axial slices:

1. The point source superquadric slice $\psi_{ps} \triangleq \{(x\boldsymbol{\xi}), (x_u^{ps}\boldsymbol{\xi}_u^{ps}), (x_v^{ps}\boldsymbol{\xi}_v^{ps})\}$
2. The superquadric of confusion slice $\psi_{sc} \triangleq \{(x\boldsymbol{\xi}), (x_u^{sc}\boldsymbol{\xi}_u^{sc}), (x_v^{sc}\boldsymbol{\xi}_v^{sc})\}$

The final complete differential wavefront estimator becomes:

$$\psi = \psi_{ps} \cup \psi_{sc} = \{(x\boldsymbol{\xi}), (x_u^{ps}\boldsymbol{\xi}_u^{ps}), (x_v^{ps}\boldsymbol{\xi}_v^{ps}), (x_u^{sc}\boldsymbol{\xi}_u^{sc}), (x_v^{sc}\boldsymbol{\xi}_v^{sc})\} \tag{7}$$

Approximating the solid angle subtended by the source as a super-quadric is not in general satisfactory, except for sources which are themselves superquadrics. To handle polygonal sources, the angle subtended must be represented as a collection of such slices. Our current implementation assumes spherical luminaires.

When ψ is incident on a surface the superconic within which the power is to be deposited is determined by examining both superconic topologies resulting from the point source slice and the confusion slice.

If the superquadric of confusion estimate is used exclusively for area sources, the result will be noisy illumination detail in regions approaching singularity, due to the inadequacy of the point source sampling rate. An example of this is depicted in Colour Plate 1(a). Therefore, when this estimate, $\hat{E}_{sc}$, becomes smaller than that of the point source sample estimate, $\hat{E}_{ps}$, we revert to the point source sample, which is designed to provide an effective sampling strategy when the sample rate is low: $\hat{E} = max(\hat{E}_{ps}, \hat{E}_{sc})$.

If the BRDF of the surface is complex, we can subdivide the superquadric of confusion or even extend it. For example, a point source illuminating a directional diffuse surface [HTSG91], would result in the superquadric of confusion being extended to diffuse the incident light about the directional diffuse direction. Colour Plate 1(b) illustrates the effect of increasing the light source size.

5 Adapting to Illumination Events

We shoot rays by choosing directions within the solid angle through which the light source radiates. Directions are chosen for spherical sources using a stratified sampling approach[4]. $\psi = (x, \xi)$, x =luminaire center.

$$\xi = \begin{pmatrix} \cos\theta \sin\phi \\ \sin\theta \sin\phi \\ \cos\phi \end{pmatrix}$$

$\theta = \cos^{-1}(2u - 1)$, $\phi = 2\pi v$ and $u, v = rand(0, 1)$. As with any sampling algorithm, the sampling rate is crucial. With a sampling rate that is too coarse we will miss such effects as

[4] We assume a diffuse distribution for the sources.

- The D^0 discontinuities[5] associated with shadows cast by point sources.
- The D^0 discontinuities associated with shadows cast by touching surfaces.
- The singularities at penumbral foci, near the D^0 discontinuities.
- Singularities caused by concave mirrored surfaces and refracting media.

We can also introduce unwanted artifacts including shadow/caustics leaks and darkened regions at shared edges. Rather than simply increase the sample rate everywhere, we look for illumination events and determine whether they represent just cause for adaptively sampling in those sampling regions. An illumination event can be signalled either by wavefront convergence indicating the possibility of a singularity or when the surface intersected by the ray differs from that of its neighbours.

When tracking the slices through the environment, we may have applied many power kernels before an event is detected. In such cases we would be required to *un-shoot* the power associated with these slices before super-sampling the slice. To cater for this, we defer the application of the kernels to the maps until the entire ray tree has been traversed, storing information regarding the required depositions in a *splat list*. If no event is detected, the splat list is used to apply the deferred illumination deposits. If an event is detected however, we throw away that part of the splat list which is to be re-evaluated at a higher resolution and supersample that sub-set of the ray tree. Figure 4 shows the results of the adaptive sampling schemes for the illumination reulting from a metal band.

To determine the resolution of the new sub-division scheme, we examine the size of the kernels being deposited at the event region, and subdivide such that the support of the kernel is approximately the size of an illumination pixel. Thus we never subdivide beyond the level of detail that the illumination map is capable of displaying. Colour Plates 2 and 3 demonstrate the improvement in image quality when using adaptive sampling techniques.

5.1 The 'Feeler Ray' Pass

Due to the expense of tracking a large number of rays through the entire spherical solid angle, we use an idea from [Shi90] where an initial low resolution pass is used to simply identify the directions in which objects are visible to the light source. We fully sample within these sample regions only.

6 Results

One of the most significant results of the algorithm is that for a given surface visible to the viewer, the number of shadow rays cast to that surface, n_s, will usually be $\ll$ number of pixels, n_p, associated with the surface on the screen. This is in contrast to the 'gather' approach monte carlo systems, which typically cast $m \times n_p$ shadow rays, where typically $m = \{4 \ldots 64\}$. This is primarily due to the coherence that can be exploited by a 'shooting' monte carlo approach.

In order to increase the speed of the algorithm, we have approximated the elliptical kernels with circular kernels of radius $r = max(r_1, r_2)$. This has the

[5] A D^n discontinuity (after notation in [LTG92]) exists at x if the function is C^{n-1} continuous at x but not C^n.

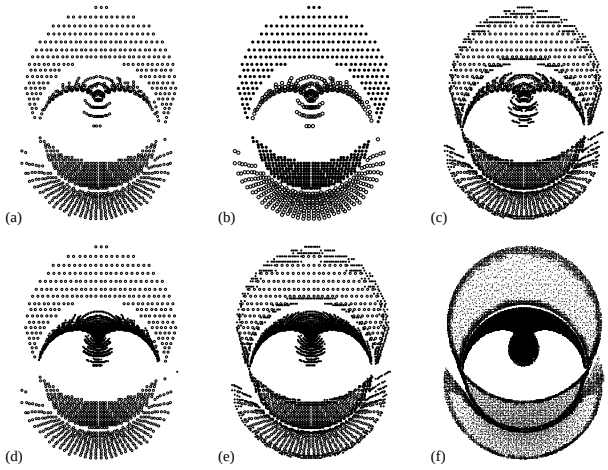

Fig. 4. Adaptive sampling schemes: (a) An initial low resolution spot diagram for the metal band in Colour Plate 4. (b) Rays classification; black spots are shadow rays, and white, the caustic rays. (c) Black spots represent rays shot as a result of adapting to illumination edges, (d) they represent rays sent to regions of illumination convergence, (e) shows the combined result and finally (f) represents a high resolution sampling, showing the concentration of samples in regions of interest.

effect of reducing slightly the illumination detail, but has the advantage that all kernels are fully symmetric, which improves the performance of the splatting process.

Colour Plate 5 demonstrates the high illumination detail that can be achieved using this algorithm. Note that the crystal object is approximated using triangular patches, and the illumination cast effectively represents a differentiation of the piecewise linear surface amplifying the edges of the patches in a manner similar to an emboss filter. The lightsource in this case has a very small radius (and is thus barely visible) and is located in the centre of the piece.

7 Conclusions

We have presented an algorithm to determine the illumination from area luminaires which captures both low and high frequency detail in the illumination using density-estimation and adpative sampling techniques, representing both the differential wavefront area around power carrying rays and the solid angle subtended by the light source as a superquadric slice, defined by an axial ray and 2 pairs of paraxial rays.

Problems remain in the current implementation. In order to be truly useful arbitrary luminaires should be allowed (our current implementation is restricted to spherical geometries). This in principle should be a straightforward extension to the system, allowing super-sampling of the superquadric of confusion. The storage of the illumination maps currently represents the major memory bottle-

neck in the system. The use of a wavelet basis [Chu92] to encode the illumination map is under investigation and current results are very promising.

8 Acknowledgements

Thanks to Dr. Dan McCarthy, my supervisor, Prof. John Byrne for his continued support and funding of this project and to the members of the Image Synthesis Group for helpful suggestions and critique.

References

[Arv86] Arvo J., Backward Ray Tracing, *SIGGRAPH '86 Developments in Ray Tracing seminar notes*, Vol. 12, Aug. 1986.

[Ase92] Asensio F., A Hierarchical Ray-Casting Algorithm for Radiosity Shadows, *Proceedings of the 3rd Eurographics Workshop on Rendering*, pp:179-188, May 1992.

[Bar86] Barr A.H., Ray Tracing Deformed Surfaces, *Computer Graphics*, 20(4):287-296, August 1986, ACM Siggraph '86 Conference Proceedings.

[Chu92] Chui C.K., *An Introduction to Wavelets*, Vol.1 of *Wavelet Analysis and its Applications*, Academic Press, Inc. Boston, MA, 1992.

[Col94] Collins S., Adaptive Splatting for Specular to Diffuse Light Transport, *Proceedings of the 5th Eurographics Workshop on Rendering*, pp:119-135, June 1994.

[Got93] Gotsman C., Constant Time Filtering by Singular Value Decomposition, *Proceedings of the 4th Eurographics Workshop on Rendering*, pp:145-155, June 1993.

[HTSG91] He X.D., Torrance K.E., Sillion F.X., Greenberg D.P., A Comprehensive Physical Model for Light Reflection, *Computer Graphics*, 25(4):175-186, July 1991, ACM Siggraph '91 Conference Proceedings.

[Hect87] Hecht H., *Optics 2nd Ed.*, pp:128-240, Addison Wesley, 1987.

[Hec90] Heckbert P.S., Adaptive Radiosity Textures for Bidirectional Ray Tracing, *Computer Graphics*, 24(4):145-154, August 1990, ACM Siggraph '90 Conference Proceedings.

[LTG92] Lischinski D., Tampieri F., Greenberg D.P., Discontinuity Meshing for accurate Radiosity, *IEEE Computer graphics and Applications*, 12(6):25-39, November 1992.

[STN87] Shinya M., Takahashi T., Naito S., Principles and Applications of Pencil Tracing, *Computer Graphics*, 21(4):45-54, July 1987, ACM Siggraph '87 Conference Proceedings.

[Shi90] Shirley P., A Ray Tracing Method for Illumination Calculation in Diffuse-Specular Scenes, *Proceedings of Graphics Interface*, Canadian Information Processing Society, pp:205-212, Toronto, May 1990.

[Shi95] Shirley P., Wade B., Hubbard P.M., Zareski D., Walter B., Greenberg D.P., Global Illumination via Density Estimation, *Proceedings of the 6th Eurographics Workshop on Rendering*, June 1995.

[Sil86] Silverman B.W., *Density Estimation for Statistics and Data Analysis*, Chapman and Hall, London 1986.

[War92] Ward G., The RADIANCE Lighting Simulation System, *Global Illumination*, ACM Siggraph '92 Course Notes.