Rendering Crystal Glass

Steven Collins[○]

Trinity College Dublin, Ireland.

1. Abstract

Rendering crystal glass presents particular problems due to the inadequate illumination models employed by the majority of rendering systems. In particular the problems of dealing with the wavelength dependent refraction of light as it passes through glass of high *dispersive power* and the patterns of focussed light cast by the glass are problems that must be addressed in order to achieve an acceptable level of realism.

To capture the afore-mentioned phenomena a 2-pass ray-tracing algorithm is used, with an adaptive light-pass followed by a standard eye-pass. During the light-pass, rays are traced from the light sources (essentially sampling the wavefront radiating from the sources), each carrying a fraction of the total power per wavelength of the source. The interactions of these rays with diffuse surfaces are recorded in ‘illumination-maps’, as first proposed by Arvo [Arvo86]. The key to reconstructing the intensity gradients due to this light-pass lies in the construction of the illumination maps. We record the power carried by the ray as a ‘splat’ of energy flux, deposited on the surface using a gaussian distribution kernel. The kernel of the splat is adaptively scaled according to an estimation of the wavefront divergence or convergence, thus resolving sharp intensity gradients in regions of high wavefront convergence and smooth gradients in areas of divergence.

The 2nd pass eye-trace (an adapted ray tracer) modulates the surface’s radiance according to the power stored in the illumination map in order to include the specular to diffuse light modelled during the first pass. This 2nd pass also takes into account the wavelength dependent refraction of light passing through glass, stochastically samples the visible spectrum, and using quadrature rules, reconstructs the spectra for final display.

[○] The author can be contacted at Steven.Collins@cs.tcd.ie. All colour images in this paper can also be obtained by http using this link: http://vangogh.cs.tcd.ie/scollins/work.html

2. Introduction

Ultimately, to produce realistic renderings of a scene, we wish to determine the radiance at every point in the scene visible to the viewer. The radiance at any point in the scene may be expressed using a recursive integral:

$$L(x,\theta,\phi,\lambda) = L_e(x,\theta,\phi,\lambda) + \int_\Omega \rho_{bd}(\theta,\phi,\theta',\phi',\lambda) L(x,\theta',\phi',\lambda)\cos\theta d\omega \qquad (2.1)$$

$L(x,\theta,\phi,\lambda)$ is the radiance (energy per unit time per unit projected area per unit solid angle per wavelength) leaving point x in direction (θ,ϕ) and is defined in terms of the radiance emitted by the surface in that direction, $L_e(x,\theta,\phi,\lambda)$ and the radiance impinging on point x from all other directions (θ',ϕ') scaled by the *bi-directional reflectance distribution function* (BRDF) which is a function of the incoming and outgoing angles and the wavelength. Existing algorithms, ray tracing and radiosity solve for many of the modes of light transport but not all. The most notable omission is that of specular to diffuse light transport, or light that reaches a surface through one or more specular interactions (either reflection or refraction).

The technique presented here implements a 2-pass method first proposed by Arvo [Arvo86] and later augmented by Heckbert [Heckbert90] to model the specular to diffuse transport as a first pass, tracing rays from the light sources as they interact with specular surfaces and depositing power on these surfaces if they exhibit diffuse characteristics and terminating at a purely diffuse surface (or when the power falls below and pre-defined threshold). A subsequent eye-pass backwards traces paths from the eye and extracts the power deposited by the first pass onto the diffuse surfaces.

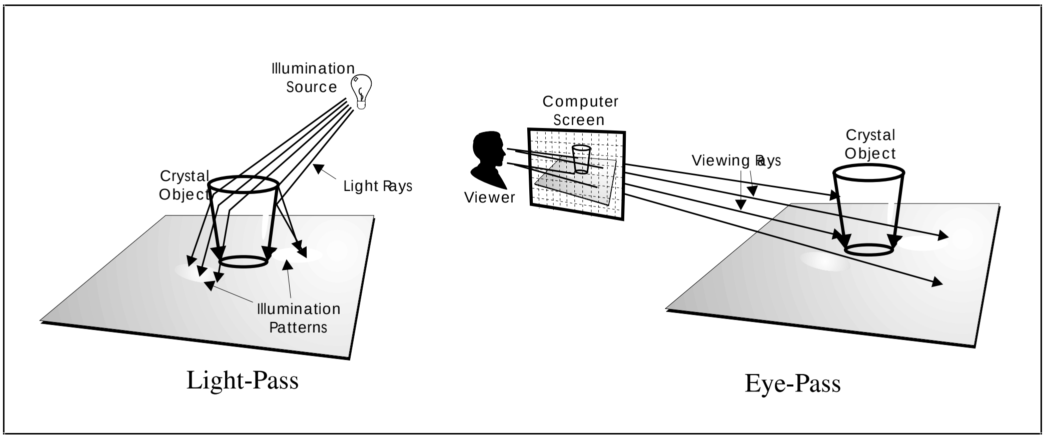

Figure 1: Illustrating the 2 phases of the algorithm

3. Tracing Specular to Diffuse Paths

The task of tracing these paths is a 3-fold operation:

1. Trace rays from the light sources, assigning a fraction of the total light source power Φ_s to each ray.
2. Record the rays collisions with diffuse surfaces by depositing this power, Φ_r, on the surface.
3. Reconstruct the power distribution during the eye-pass by modulating the radiance from the surface according to the power stored on the surface.

We will now discuss the details of each of these phases and their implementation in the current system.

3.1 Light Ray Tracing

We trace rays originating at the light sources in the scene and record the interactions of these rays and the surfaces in the scene, depositing power on diffuse surfaces along the rays' paths. In order to record this power, *illumination maps* (in the spirit of Arvo [Arvo86]) are used. For details of the algorithm see [Collins94].

3.1.1 Light Ray Distribution

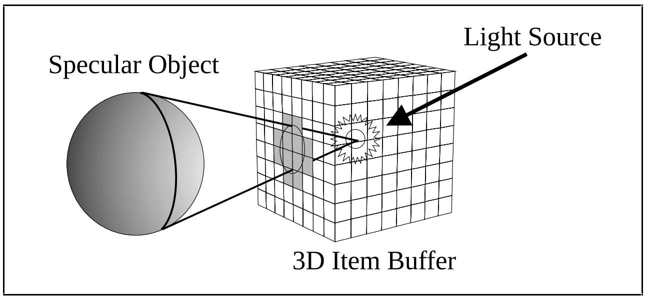

Figure 2: Item Buffer Construction.

In order to determine the solid angle, about the light sources within the scene, which must be sampled, we use a technique similar to the item buffer of [Weghorst84] and the hemicube of [Cohen85]. We wish to trace light-rays only in directions that will potentially give rise to some from of specular to diffuse transfer, so we first determine the directions in which specular surfaces are visible from the light source. All the objects in the scene are projected onto a unit cube centred at the light source with Z-buffers attached to its faces. During the light pass, we sample only those 'pixels' on the cube's faces that can 'see' a specular surface. The number of rays sent to each cube pixel is determined by the solid angle subtended by the pixel and a user specified sample density (specified in rays per steradian).

Each ray carries with it a fraction of the total power of the source. This fraction is proportional to the solid angle subtended by the ray at the source. The power shot towards each cube-pixel for a point light source is:

$$\Phi_p = \frac{\Phi_s \Delta\omega}{\omega_s} = \frac{\Phi_s \Delta\omega}{4\pi} \qquad (4.2)$$

3.2 Wavefront Behaviour Tracking

We model the behaviour of the wavefront emanating from the light source by examining the distribution of ray hits across surfaces. On a given surface we are interested in the convergence/divergence of the wavefront locally. Rather than attempt to reconstruct the shape of the wavefront by applying a post-filter to the intersection points we maintain connectivity information between rays. By tracking arrays of rays from the light source, we associate with each ray some fraction of the solid angle of the source. When these rays intersect surfaces we examine the distances between neighbouring rays to determine the shape of the wavefront. We use this information to determine how to distribute the power on the receiving surface.

3.2.1 Deposit Area Determination

We maintain neighbourhood information between rays through the use of a dynamic caching system for each surface where the most recent set of rays and their collisions with the surfaces are recorded, thus for any new ray colliding with a surface, we can estimate local wavefront density by examining the last few ray hits. If enough information does not exist to determine the spread, deposition of power is postponed until new rays provide the required information.

Figure 3 depicts a subset of the rays from a light source incident on a surface, having passed through the scene, possibly interacting with specular surfaces. Only when ray number 4 hits the surface do we have enough information to estimate the spread of the wavefront. Rays 1 to 3 are cached until 4 arrives, and the estimated spread is used for all 4 deposits (as there will not be enough information to independently determine an estimate for rays 1 to 3). The spread estimate is simply the area of the polygon defined by the 4 ray hit points. This area is effectively the area over which we wish to deposit the power being carried by the ray.

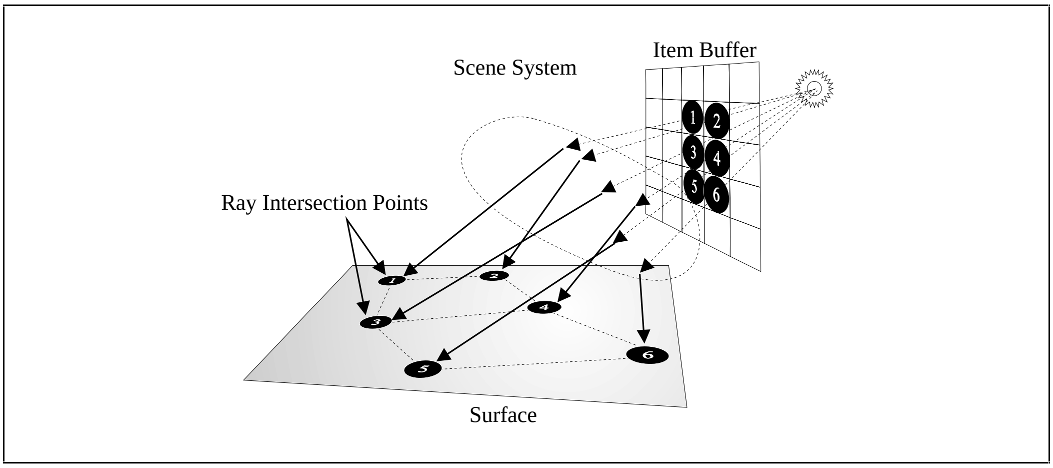

Figure 3: Determine Wavefront Spread from Ray Intersections.

In order to determine an accurate representation of the wavefront, we must be careful in how we correlate rays and their intersections. We only wish to match rays that have arrived via the same transport mechanism (in other words, rays that are at the same point in the ray-tree). If we try to correlate a ray that has been reflected off a surface with a ray that has been refracted, the result will be an incorrect estimate of the wavefront as we are dealing with different wavefront *time-steps*. To achieve correct correlation, the dynamic cache of ray hit points is a tree matching the ray-tree itself, allowing us to match rays at the same level in the ray tree. For more details see [Collins94b].

3.2.2 Power Deposition using Gaussian Kernels

Having determined the area over which we wish to deposit the power, we must now decide how to distribute this power across that area. Rather than distribute the power uniformly across the area which tends to result in aliasing at the edges of the regions (see [Watt90]) we deposit the power as a *splat*. This technique has been used to implement a progressive refinement algorithm for volume rendering [Laur91] where gaussian splats are used to represent the volume's octtree and the footprint of the splats scaled to match the projected area of the voxels making up the octtree.

We deposit power, therefore, with a standard gaussian distribution:

$$w(x,y) = e^{-\frac{z^2}{2\sigma}} \tag{4.4}$$

The footprint of the gaussian kernel is scaled to match the areas defined by the ray collisions.

3.3 Shadows

As a useful by-product of the wavefront tracking algorithm, it is a trivial extension to track the *shadow wavefront*. Rather than shooting light power with each ray, we shoot *negative light* power, and using the same wavefront divergence/convergence estimates can accurately reconstruct the shadows cast by the objects. Details of this algorithm can be found in [Collins94a].

3.4 Eye pass

During the eye pass, the illumination map is queried in a similar fashion to a texture map (though, unlike texture maps, it is not subject to shadowing). The power in each pixel of the illumination map is converted to intensity making the assumption that the surface is perfectly diffuse:

$$\mathrm{L}(\mathrm{x},\mathrm{y},\lambda) = \frac{\Phi_{(\mathrm{x},\mathrm{y})}}{\pi \mathrm{A}_{(\mathrm{x},\mathrm{y})}} \tag{4.5}$$

Φ_p is the power accumulated in pixel (x,y) of the illumination map and A_p is the area of the pixel in world-space.

4. Spectral Considerations

Due to the potentially high intensity artefacts resulting from wavefront convergence through transmitting media, the wavelength dependency of the refractive index can be omitted in order to render realistic images.

Wavelength dependent reflection is taken into account in the Fresnel approximation presented by Cook et al. [Cook82], which approached the problem using a geometric optics model, where reflectance is a function of both wavelength and the angle of incidence of the incoming light. The model presented by He et al. [He91] is an extension of the Cook Torrance model, based on physical optics, whereby the reflectance of a surface is dependent on the wavelength, angle of incidence, surface roughness parameters and surface refractive index.

4.1 Spectral Sampling

Spectral sampling is required for both phases of the algorithm; first during the specular-to-diffuse phase, where the spectrum must be sampled in order to achieve the splitting of light as it passes though transparent objects such as the prism, and second during the final phase, to model the dispersive effects commonly seen looking at crystal glass (it is this effect that gives cut crystal its bright colourings).

The choice of sample points (*n*) is not clear. Musgrave, in his modelling of the rainbow, [Musgrave89], uses 13 samples, spread uniformly across the visible spectrum. Meyer [Meyer88], having examined a number of different sampling schemes, derived a new colour space, AC_1C_2, and using gaussian quadrature techniques to integrate low order sampling polynomials applied to the spectral curve, samples into this new colour space. Having tried a number of sampling schemes, Meyer suggests a scheme where 4 samples are used.

For the sake of computational efficiency, we have opted for 6 sample points. Colour Plates 3 and 4 show the effects of different spectral sampling resolutions. The test scene shows a prism and sphere of silicate flint glass. The prism is illuminated by a thin beam of white light which is dispersed to produce the spectrum on the back wall. Colour Plate 3 details the results of 2 sampling schemes. The top 6 spectra result from initial rays hitting the prism being refracted, generating a number of new rays distributed (jittered) through the visible spectrum. The bottom 6 spectra detail a ‘fairer’ scheme where each initial ray spawns only 1 refracted ray, the wavelength of which is determined stochastically.

All computations are performed with respect to these 6 sample points. For final image display however, we must convert to RGB space. This is achieved [Hall89][Peercy93] by multiplying the reconstructed spectrum by the CIEXYZ tristimulus matching curves, using Riemann summation to evaluate the integral, and finally converting the XYZ colour to RGB space using chromaticity data for the monitor phosphors. Higher order quadratures such as Gauss-Legendre would improve the recontruction but at a cost of higher computation times. Riemann summation with *n* sample points has been shown to be exact for the first *2n+2* fourier functions. If we assume that spectral curves can be approximated by these functions (which is generally the case in natural lighting) then this approximation will suffice.

4.2 Material Data

The *mean dispersion* $(\eta_F - \eta_C)$ of a material defines the change of index of refraction of that material with respect to frequency and the following ratio is used to define the dispersive behaviour of a material (usually glass):

$$v = \frac{\eta_d - 1}{\eta_F - \eta_C} \tag{5.2}$$

η_F, η_C are the refractive indices of the material at the F and C Fraunhofer lines (the emission lines of hydrogen at 486.1nm(blue) and 656.3nm(red) respectively) and η_d the refractive index at the yellow d line of helium (587.6nm) rather than the sodium D doublet (589.3nm) used in older literature. This ratio is *Abbe's number*, also called the *v value* or *V-number* of the material. A high V-number indicates a material of low mean dispersion. Manufacturers of glass classify the glass by its glass number, a six digit code, the first three digits of which is the refractive index $\eta_d - 1$ and the last three digits Abbe's number scaled by 10. For example a glass with number 523588 has a refractive index of η_d = 1.523 and mean dispersion of 58.8 [Meyer-Arendt89]. Glasses with low mean dispersion (V-number above 55) are called crowns and glasses of high dispersion (V-number below 50) are called flints.

Using this information we can reconstruct a curve relating frequency and refractive index, and sample this curve at our spectral sample points to determine the index to use for our light-rays.

Another approach is to adopt one of the many attempts to formulate a quantitative relationship between refractive index and frequency such as that of Herzberger [Herzberger59]:

$$\eta = A + B\lambda^2 + \frac{C}{\lambda^2 - \lambda_0^2} + \frac{D}{(\lambda^2 - \lambda_0^2)^2} \tag{5.3}$$

A, B, C and D are empirically determined constants, and λ_0 is set at 168nm. Values for A, B, C, and D are difficult to come by and therefore the glass number is a more convenient representation. The glass used in the example scenes included in this paper has been modelled on Silicate Flint Glass with glass number 635577.

5. Results

To demonstrate the application of the system on complex objects, a classic example of caustic formation, the swimming pool floor, was rendered following the example of [Watt90]. Colour Plate 1 is a view from above, and Colour Plate 2 a view from below the surface of the water. As remarked by Watt, the triangular patch resolution of the water's surface is amplified by the caustic image, requiring very high patch subdivision resolutions. To avoid this we modelled the surface as a single polygon having a bump map composed of overlapping cosine terms [Max81].

Colour Plate 3 shows 3 instances of crystal glass shapes. Note the complex patterns of focussed light cast be each of the 3 pieces. Colour Plate 4, a prism, and 5, a large diamond, depict the effects of the spectral sampling employed during the 2nd phase of the algorithm. Note the coloured edges in the images seen through the glass prism and the diamond due to the refractive qualities of the materials.

6. Conclusions

We have presented an algorithm extending existing specular to diffuse transport models with a view to capturing both high and low frequency artefacts through the use of adaptive gaussian kernels. We also extend the method to allow for accurate modelling of frequency dependent refraction of light with a view to capturing the beauty of dispersion, a particularly important feature when rendering crystal glass.

The illumination maps are typically very expensive in terms of memory usage, despite the adaptive strategy adopted here. The compression of illumination maps using wavelets or fourier compression is currently under investigation. Early results are very promising with potentially 50-70% memory savings.

7. Acknowledgements

I wish to thank my supervisor, Dr. Dan McCarthy for his continuous advice and support, Prof. John Byrne for continuing to support and fund this work, my colleagues in the Image Synthesis Group, Trinity College Dublinand finally Dr. Adrian Moore and his colleagues in the University of Ulster Coleraine for organising this 2nd Irish Workshop on Computer Graphics.

8. References

[Arvo86] Arvo J., “Backward Ray Tracing”, *SIGGRAPH ‘86 Developments in Ray Tracing seminar notes,* Vol. 12, Aug. 1986.

[Cohen85] Cohen M.F., Greenberg D.P., “The Hemi-Cube; A Radiosity Solution for Complex Environments”, *Computer Graphics,* Vol. 19, No. 3, pp. 31-40, July 1985.

[Collins94] Collins S., “Adaptive Splatting for Specular to Diffuse Light Transport”, *Proceedings of the 5th Eurographics Workshop on Rendering,* pp. 119-135, June 1994.

[Collins94a] Collins S., “Wavefront Tracking as an Inexpensive Shadow Generation Technique”, *Trinity College Computer Science Technical Report*, November 1994.

[Collins94b] Collins S., “Ray Correlation During Wavefront Tracking”, *Trinity College Computer Science Technical Report*, November 1994.

[Cook82] Cook R.L., Torrance K.E., “A Reflectance Model for Computer Graphics”, *ACM Transactions on Graphics,* Vol. 1, No. 1, pp. 7-24, 1982.

[Green86] Green N., Heckbert P.S., "Creating Raster Omnimax Images from Multiple Perspective Views using the Elliptical Weighted Average Filter", *IEEE Computer Graphics and Applications,* Vol. 6, No. 11, pp. 21-29, 1986.

[Hall89] Hall R., *Illumination and Color in Computer Generated Imagery,* Monographs In Visual Communication, Springer Verlag, pp. 45-62, 1989.

[He91] He X.D., Torrance K.E., Sillion F.X., Greenberg D.P., "A Comprehensive Physical Model for Light Reflection", *Computer Graphics,* Vol. 25, No. 4, pp. 175-186, July 1991.

[Heckbert90] Heckbert P., "Adaptive Radiosity Textures for Bidirectional Ray Tracing", *Computer Graphics,* Vol. 25, No. 4, pp. 145-154, August 1990.

[Herzberger59] Herzberger M., "Colour Correction in Optical Systems and a New Dispersion Formula", *Opt. Acta (London),* Vol. 6, pp. 197-215, 1959.

[Laur91] Laur D., Hanrahan P., "Hierarchical Splatting: A Progressive Refinement Algorithm for Volume Rendering", *Computer Graphics,* Vol. 25, No. 4, pp. 285-288, July 1991.

[Max81] Max N.L., "Vectorised Procedural Models for Natural Terrain: Waves and Islands in the Sunset", *Computer Graphics,* Vol 15, No. 3, pp 317-324, August 1981.

[Meyer88] Meyer G.W., "Wavelength Selection for Synthetic Image Generation", *Computer Vision, Graphics & Image Processing,* Vol. 41, pp 57-79, 1988.

[Meyer-Arendt89] Meyer-Arendt J.R., *Introduction to Classical and Modern Optics*, Third Edition, Prentice-Hall International, pp. 13-26, 1989.

[Musgrave89] Musgrave F.K., "Prisms and Rainbows: a Dispersion Model for Computer Graphics", *Graphics Interface '89,* pp. 227-234, 1989.

[Peercy93] Peercy M.S., "Linear Color Representations for Full Spectral Rendering", *Computer Graphics Proceedings, Annual Conference Series 1993,* pp. 191-198, August 1993.

[Watt90] Watt M., "Light Water Interaction using Backward Beam Tracing", *Computer Graphics,* Vol. 24, No. 4, pp. 377-385, August 1990.

[Weghorst84] Weghorst H., Hooper G., Greenberg D.P., "Improved Comutational Methods for Ray Tracing", *ACM Transactions on Graphics*, Vol. 3, No. 1, pp. 52-69, January 1984.