Spectrum of Glass Rendering

Jacy Scharlow

Abstract – Glass is an allusive material when it comes to rendering how it interacts with the environment around it, but when properly simulated can create a visually compelling shot. Glass objects can have a big impact on how the scene is rendered in its entirety but being able to control different qualities of a glass material is limited.

This paper discusses the model for glass material representation, and techniques on how a program can use these material descriptions to render the desired image specifically using ray tracing and photon mapping. This paper will cover the phenomena: refraction, colored transmission, caustics, texture, and dispersion.

1 INTRODUCTION

Glass has a wide range of qualities to it that can be explored when experimenting with materials that can be put into a scene. Glass can be completely clear while also having qualities of specular reflections. It can be stained and frosted. Certain types of glass can act as a prism and disperse white light into a spectrum of color. The versatile nature of this material results in glass becoming a focal point in architecture and photography, but due to some of these feature's glass is often cheated in the digital medium.

In this paper we explore ways to represent these variables in object material, so that we can then render glass in a way that could mimic reality. This provides a complete explanation of variables and necessary precomputations needed to obtain a reasonable result when rendering these objects. We will discuss practical issues regarding these techniques, and why it may be easy to get a flawed result. From there we will explain how raytracing and photon mapping uses the variables to represent these qualities in a rendered scene, providing the equations and discussions on the logic.

We also discuss time runtime and efficiency of these algorithms as we look back on the techniques used in this research. Furthering our discussion into future works and improvements that can be implemented to make these solutions increasingly more feasible to explore.

Next, we will discuss related work, and then proper refraction, reflection, and diffusion in Section 3, colored transmission and projections in Section 4, rough surface simulation in Section 5, and how we should handle dispersion in Section 6. From there we will discuss the quality of data structure in Section 7, and finally conclusion and future work in Section 8.

2 RELATED WORK

The ray tracing models used in this assignment were implemented from the model introduced by Whitted [7], and the photon mapping to render global illumination and caustics were explored by Jensen [2] [3].

Colored transmission and caustics through general transparent materials were discussed by McGuire [4], and also discussed partial visibility in all thing transparent not just glass.

Rough surfaces on glass were demonstrated by Walter et al. [6] and de Rousiers et al. [1]. These papers used distribution functions to help simulate roughness in BRDF and BTDF.

3 REFLECTION, REFRACTION, AND DIFFUSION

Variable	Definition
d	Diffuse color vector
r	Reflection color vector
t	Refraction color vector
σ	Refraction index

***Color Vector - <Red, Green, Blue>

3.1 Diffusion

Diffusion of light is typically represented in the scattering of light. Outside of graphics this is typically common in objects that have a rough surface in opposition to materials where the surface is glossier.

Diffusion in ray tracing is typically calculated along side of the global illumination (**Gi()**) of the object. This can either be approximated with an ambient light value or calculated with photon mapping.

1)

$$\mathbf{D} = \mathbf{d} \cdot \mathbf{Gi()}$$

If we calculate diffusion as is and apply it to the overall computation to an object, typically the this will cause a mask in weighted color **d** to be applied on top of the computed color of reflection and refraction, because they will be all summed together at the end.

In photon mapping, light is dispersed around the scene in random directions. Simulating the scattering of light should it hit a diffuse object.

3.2 Reflection

Reflection of light is when light is bounced off an object into a mirrored direction. In backwards raytracing this allows reflective objects to see neighboring objects and project a mirror onto its own surface.

In backwards raytracing, reflection is calculated by casting rays outward from the camera and bouncing these rays each time they hit an object until the rays reach a predetermined depth. Where the results are then summed together.

2)

$$\mathbf{R} = \mathbf{r} \cdot \mathbf{Tr}(p, \overrightarrow{d_r}, b+1)$$

In the above equation **Tr()** is the trace ray function, where it takes in p is the intersection point, and $\overrightarrow{d_r}$ is the reflected direction, and b is the bounce of this iteration of the recursive function.

3)

$$\overrightarrow{d_r} = \overrightarrow{d_\iota} - \mathbf{2}(\overrightarrow{d_\iota} \cdot \vec{n})\vec{n}$$

Here $\overrightarrow{d_\iota}$ is the incoming direction and $\vec{n}$ is the normal direction of the collided surface.

In photon mapping, reflection bounces the light off specular objects, concentrating light in different parts of the scene.

3.3 Refraction

Material	Refraction Index
Crown Glass (pure)	1.50 – 1.54
Flint Glass (pure)	1.60 – 1.62
Crown Glass (impure)	1.47 – 1.76
Flint Glass (impure)	1.52 – 1.93

Refraction is the transmission of light through an object. Translucent and transparent objects are typically produced by this calculation.

In backwards ray tracing, as rays hit a transmissive object the ray will bend and travel through the object until it exists the object, bending again while it does. Calculating refraction is done by Snell's Law:

4)

$$\eta = \frac{\eta_1}{\eta_2}$$

If the ray is cast from outside and object, then η_1 is 1 and η_2 is σ. If the ray is cast from inside an object, it is the inverse.

5)

$$c_1 = \overrightarrow{d_i} \cdot \vec{\mathrm{n}}$$

$$c_2 = \sqrt{(1-\eta^2)(1-{c_1}^2)}$$

$$\overrightarrow{d_t} = \eta \overrightarrow{d_i} + (\eta c_1 - c_2)\vec{\mathrm{n}}$$

$$\boldsymbol{T} = \boldsymbol{t} \cdot \mathbf{Tr}(p, \overrightarrow{d_t}, b)$$

After transmission is calculated, everything past the object is computed and our transmitted colored are then returned through T.

In photon mapping the photons travel through the object and projects at the calculated direction on the other side.

3.4 In Application to Glass Materials

The color vectors of the materials are important to how the program intends to weight each of the above calculations. The are specific rules to how these vectors are used:

6)

$$\mathbf{lu} = \mathbf{d} + \mathbf{r} + \mathbf{t}$$

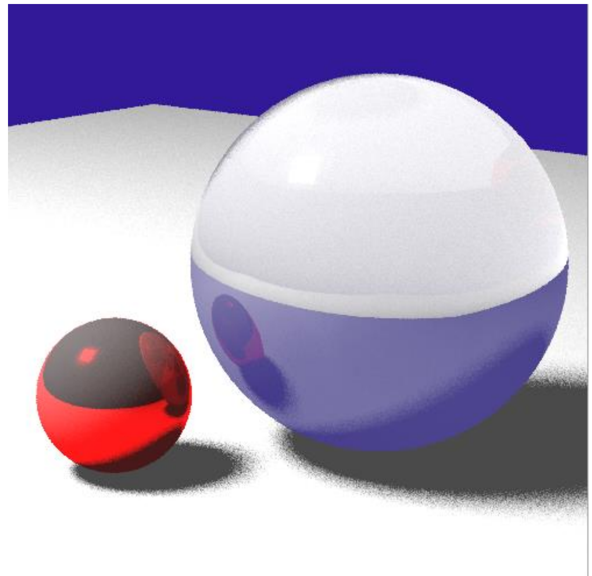


Figure 1: One glass sphere and one red specular sphere. The glass sphere has a variable balance of : Diffusion <0.1,0.1,0.1> Reflection <0.1,0.1,0.1> and Refraction <0.8,0.8,0.8>

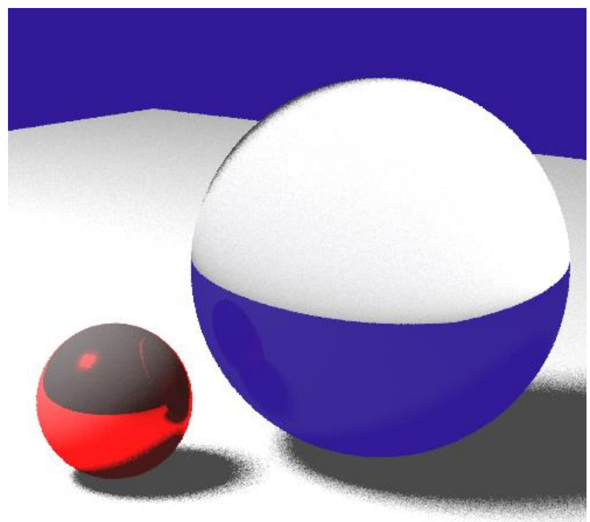


Figure 2: One glass sphere and one red specular sphere. The glass sphere has a variable balance of : Diffusion <0,0,0> Reflection <0.01,0.01,0.01> and Refraction <0.99,0.99,0.99>

lu is the light use vector, which is the sum of the three-color vector variables. Each element of the vector should be with the constraints of 0 and 1.

lu = <Red, Green, Blue>

0 >= Red <= 1

0 >= Green <= 1

0 >= Blue <= 1

The above rule is also relevant for all three of the individual vectors as well. It is note that **lu** cannot exceed 1 but it does not need to equal 1. This is because a material does not need to use all the light that it meets.

7)

$$\boldsymbol{\alpha} = <\mathbf{1}, \mathbf{1}, \mathbf{1}> - \boldsymbol{lu}$$

In the above equation, $\boldsymbol{\alpha}$ is the absorbed light dictated by the material.

Glass in general is understood as a refractive material. It would then be understood that the material color vectors (**d, r, t**) are weight in favor of the refractive variable. By weight we do not mean that the vectors are multiplicative, but instead have a higher total sum when all elements in the vector are added up. (Figure 1, 2)

However, when consider different types of glass it is important to not that sometime the light passed through is rather minuet. When making a material such as obsidian glass it is important to not its specular reflection, but also the fact that it does not transmit that much light through it either.

4 CAUSTIC AND COLORED TRANSMISSION

If a ray hit an object on the way to the light, that area is shaded according to the percentage of rays that did not hit the light. Instead, photon mapping determines the caustics of the light passing through an object. When light travels through a highly refractive object such as glass, photons may have the tendency to condense into a specific area. (Figure 3) So if photons overlap with a shadow at a high enough density, then it is as if a shadow was not cast in that area at all.

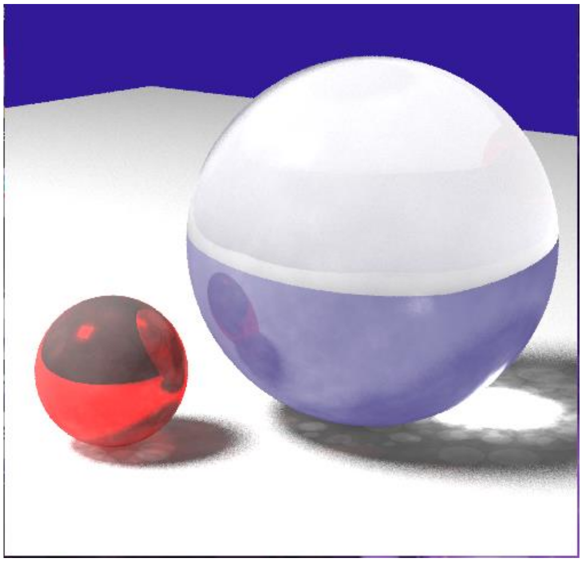


Figure 3: One glass sphere and one red specular sphere. This demonstrates the way light passes through glass objects of a high enough refraction index.

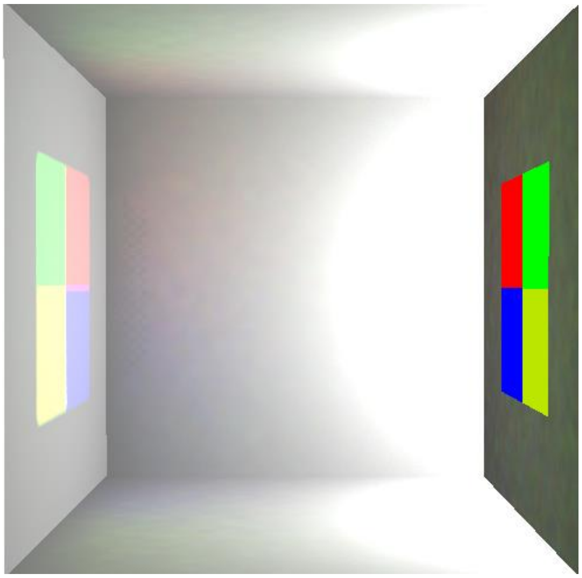


Figure 4: Transmissive material on right wall, and projected caustic result on left wall.

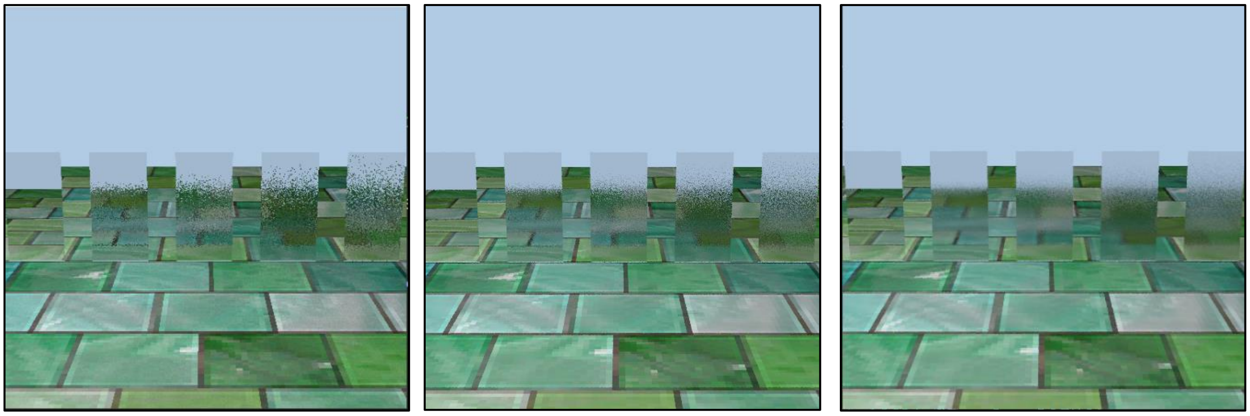


Figure 5,6,7: Left figure has no antialiasing and is only casting a single convoluted ray at an increased possible radius from left to right. Middle figure is the same as the left figure but include antialiasing. Right figure has antialiasing but is also projecting three convoluted ray per collision with the glass.

Now when referencing the material variables, there is a refractive color vector. Typically, when light shines through a refractive object, it will pick up the color of that object. When considering our photon mapping, the photons would multiply the color of the light with the refractive color of the object. If enough photons of that light's color lands on a surface after refracting through the glass, it may have a chance of projecting that color elsewhere (figure 4).

One thing to note in figure 4, this effect is heavily dependent on the type of light sources in the scene. This scene has one area light source that projects all its photons in its normal direction. Should the light source been set to generate the photons into random direction this effect would not have been possible. As a diffuse light source would have cast all the photons so that they would blend on the projected surface recreating white light.

It is interesting to note that most stained-glass scene in architecture do not have this problem, and that is because the light source of that stained glass is the sun, where all the light is coming for almost the same direction. Like the scene create here.

5 ROUGH SURFACE

Variable	Definition
d	Diffuse color vector
r	Reflection color vector
t	Refraction color vector
σ	Refraction index
λ	Roughness

Representing roughness in a glass material in key to being able to render frosted glass at various degrees of texture. When light hits a rough surface of an object, the normal of the face can be altered at varied angles. Some microfacets can be extreme, while others will may only offer a slight blurring effect.

We made an equation used to sample part of a unit sphere around a given unit vector.

8)

$$\gamma = rand[0,1] * (\lambda/2)$$

$$\vec{\rho} = \vec{\mu} - (\vec{\mu} \cdot \vec{d_\iota})\,\vec{d_\iota}$$

$$\vec{\beta} = ((p + \vec{\rho}) - (p + \vec{d_\iota}))\gamma$$

$$\overrightarrow{d_s} = \left(p + \overrightarrow{d_i} + \vec{\beta}\right) - p$$

$$\widehat{d_s} = \frac{\overrightarrow{d_s}}{|\overrightarrow{d_s}|}$$

This equation will convolute a given unit vector within a given range addressed by the material variable λ. This approach is like how antialiasing would sample a pixel, but instead of sampling a set area, we are sampling an arbitrary radius.

This affect is applied to both reflection and refraction in ray tracing and photon mapping. The calculated pixel color is the sum of all the dispersed rays, and in photon mapping a single photon gets scattered in several weaker photons upon collision with a rough material.

9)

$$\mathbf{R} = \sum_{i=0}^{n} (\mathbf{r} * \frac{1}{n}) \cdot \mathbf{Tr}(p, \widehat{d_{rs}}, b + 1)$$

$$\mathbf{T} = \sum_{i=0}^{n} (\mathbf{t} * \frac{1}{n}) \cdot \mathbf{Tr}(p, \widehat{d_{ts}}, b)$$

The more samples you take of the rays the smoother the blurred result will be. The large the roughness variable is ($0 >= \lambda <= 1$) the more widespread the diffuse effect will be.

6 DISPERSION

Variable	**Definition**
d	Diffuse color vector
r	Reflection color vector
t	Refraction color vector
σ	Refraction index
λ	Roughness
δ	Distribution

Refraction is not as simple as a single refraction index for each object's material. It is suitable for the purposes of approximating how glass works. Glass refracts light at a

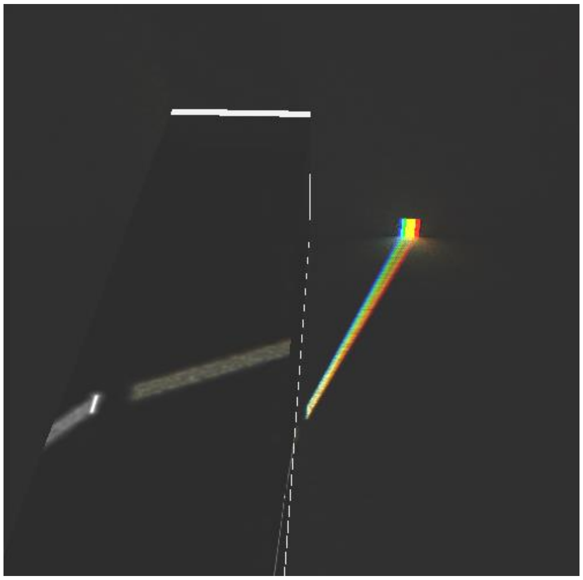


Figure 8: shows light being cast through a constructed prism, 1.6 refraction index and a distribution of around 0.032.

different rate depending on the light's wavelength or more colloquially known as the light's color.

Material	Refraction	Distribution
Crown Glass	1.5	0.01
Flint Glass	1.75	0.034

Distribution is the refraction index difference between green and blue wavelengths. Prisms are made from dense flint glass, making the dispersion more dramatic, allowing for a noticeable spectrum to appear as white light passes through it.

In order to simulate this phenomenon, we strictly accomplished this in photon mapping. If dispersion is a desired effect, the program would detect if light hitting our glass object was white (a photon with equal red, green, and blue values). The photons would then split into four different colors (red, yellow, green, and blue). The photon would then be refracted at the designated refraction index:

10)

$$ri = \sigma + (\delta * i)$$

This would then create a rainbow on the other side, and any overlapping photons should assist in creating a gradient of color. (Figure 8)

Angles and areas of light are important to getting a quality prism effect because some of the photons may just blend together if they are not spread out enough.

7 RESULTS

These techniques were developed and testing on a PC with an Intel® Core™ i7-7700HQ CPU, 32 GB of memory, and a GeForce GTX 1060 with Max-Q Design. Most of the solutions in each of these aspects of glass included creating more rays or photons. Most ambitious renders took over night if they ere using photon mapping, average photon mapping rendering required an hour.

General ray tracing renders are quite fast if there are no photons. Many renders take one to two minutes, but if there is antialiasing or roughness involved, it can take up to five minutes.

Testing of the concepts described above consisted of making new scene that was supposed to work with a set of these qualities we were working with. Then we would adjust the variables to see how they changed progressively. Most of what we attempted to emulate came from picture of real-world examples found on the internet, and we would then see if we could make a similar effect with what we managed to accomplish here.

8 CONCLUSION & FUTURE WORK

We have explored many qualities of glass, and designated solutions on how to render all of them. Not only do these render techniques work cohesively together to create custom materials for the objects, but they are also intuitive in nature.

Due to the dismal runtime and efficiency, we would like to explore some ways to improve our algorithm to assist in testing some of the techniques to help refine some of the calculations. Possibly allowing us to cast more photons between the designated four we are currently casting in dispersion.

Future works may include exploring different types of glasses to render. Since obsidian and opalite are both types of glass, but they are not uniform in texture or color. Possibly we could also look outside of glass rendering and maybe look into some other transparent material that have other interesting qualities, like aerogel, or other materials that have depth reliant color shifts and air bubbles.

We may also want to explore illuminate of backlit glass and hollow glass objects as well.

REFERENCES

[1] Charles de Rousiers, Adrien Bousseau, Kartic Subr, Nicolas Holzschuch, Ravi Ramamoorthi. Real-Time Rendering of Rough Refraction. IEEE Transactions on Visualization and Computer Graphics, Institute of Electrical and Electronics Engineers, 2012, 18 (10), pp.1591-1602.ff10.1109/TVCG.2011.282ff. ffhal-00652076f

[2] Jensen, Henrik Wann. “Global Illumination Using Photon Maps.” Eurographics, 1996, pp. 21–30.,

doi:10.1007/978-3-7091-7484-5_3.

[3] Jensen, Henrik Wann. “Rendering Caustics on Non-Lambertian Surfaces.” Computer Graphics Forum, vol. 16, no. 1, 1997, pp. 57–64., doi:10.1111/1467-8659.116.

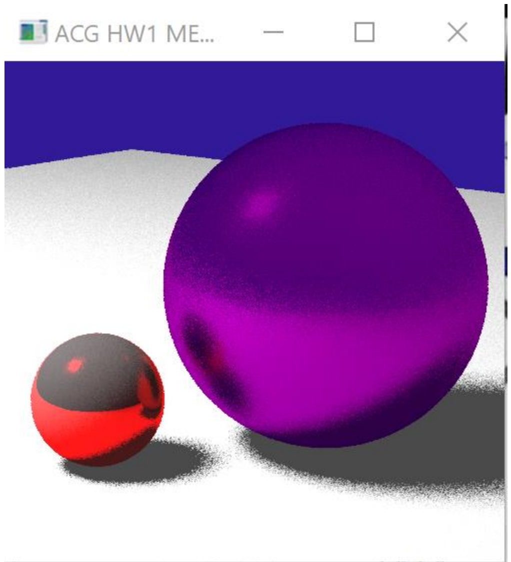
[4] McGuire, Morgan, and Michael Mara. “A Phenomenological Scattering Model for Order-Independent Transparency.” Proceedings of the 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games, 2016, doi:10.1145/2856400.2856418.

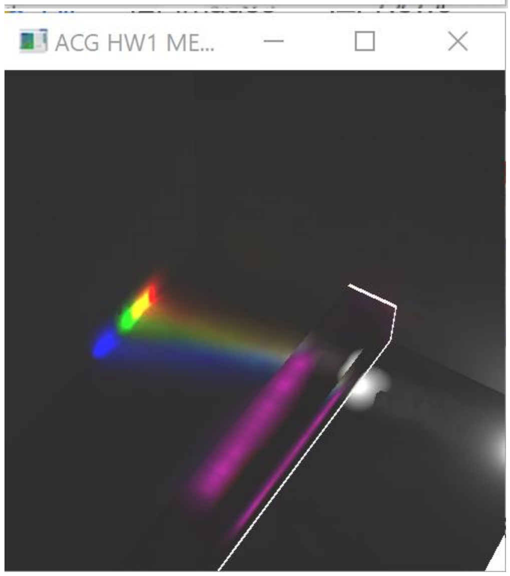
[5] Sun, Yinlong, et al. “RENDERING LIGHT DISPERSION WITH A COMPOSITE SPECTRAL MODEL.” International Conference on Color in Graphics and Image Processing, 2000.

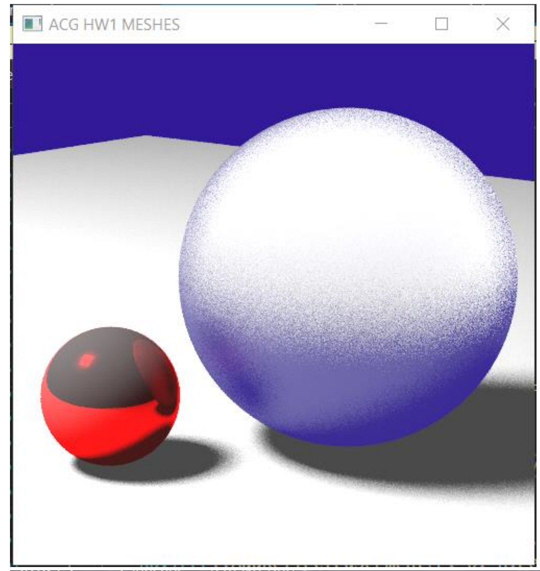
[6] Walter, Bruce, et al. “Microfacet Models for Refraction through Rough Surfaces.” Eurographics Symposium on Rendering, 2007.

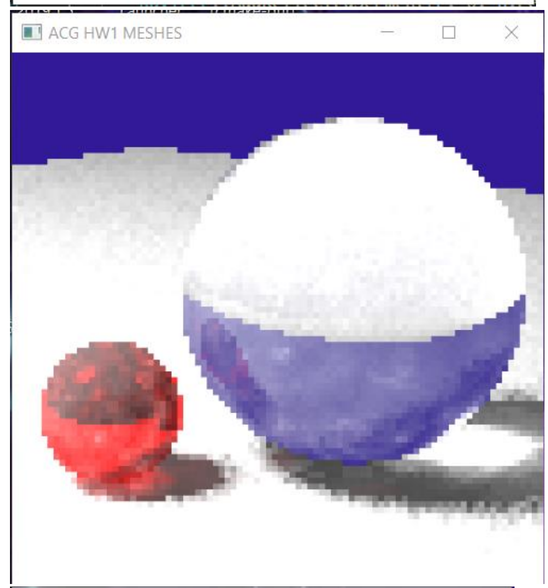
[7] Whitted, Turner. “An Improved Illumination Model for Shaded Display.” ACM SIGGRAPH 2005 Courses on - SIGGRAPH '05, 2005, doi:10.1145/1198555.1198743.

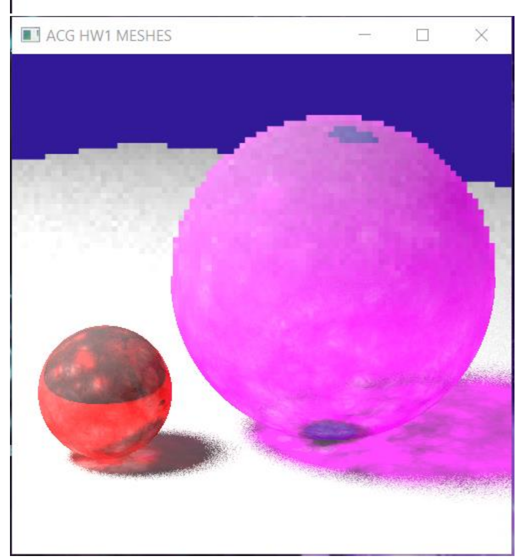
PHOTOS:

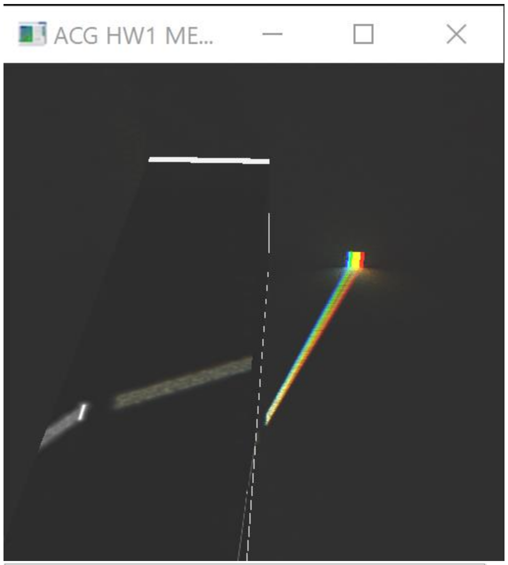

ACG HW1 ME...

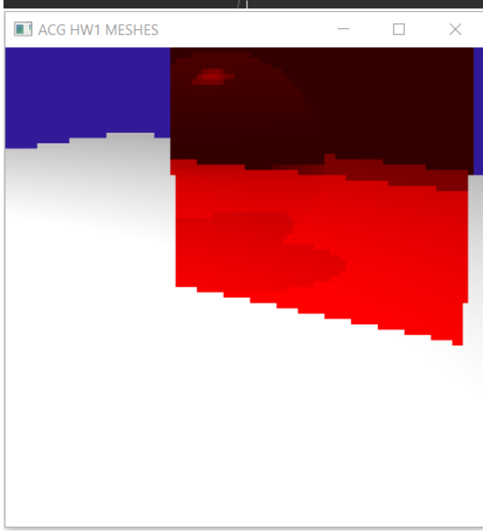

ACG HW1 ME...


ACG HW1 MESHES


ACG HW1 MESHES


ACG HW1 MESHES


ACG HW1 ME...


ACG HW1 MESHES