Rendering Candle Flames as Blackbodies

Viet Vo and Sang Goo Kang

Introduction

Our goal for this project is to render a realistic image of candle flames by utilizing the concept of blackbody radiation. We plan create a fairly dark scenery with a book, table, and candles, in which the only source of light are from the flames. The idea is the capture a time in the past without electricity where you must read and study using candle flames. Our inspiration for our project came from Image 1 and Image 2. We utilized some of the technical ideas from Pegoraro and Parker "Physically-Based Realistic Fire Rendering" in our project for our fire rendering. We represented our candle flames as a medium with zero scattering because flames have low albedo. In addition, we converted our flame spectra to the XYZ color space defined by the Commission Internationale de l'Eclairage (CIE) before converting to RGB for use in PBRT [1]. Big differences between our work and Pegoraro and Parker's work is that their fire is turbulent and takes into account molecular physics to compute spectra based on the chemical compounds found in fire.

Image 1: Photo Credits -- ShutterStock

Image 2: Photo Credits --http://byjus.com/chemistry/candle-flame/

To simplify our representation of the flames but still maintain an accurate visual representation that obeys the physics of light, we referenced Richard E. Barrans of the PG Research Foundation, Darien, Illinois on their discussion of the different layers of a candle flame. We worked on 4 out of the 5 layers of the candle flame as seen in the image below since the 5th layer (known as the veil) is not visible to the human eye. Our flame models the blue zone, non-luminous zone, dark zone, and luminous zone, all of which have different colors and temperatures [3].

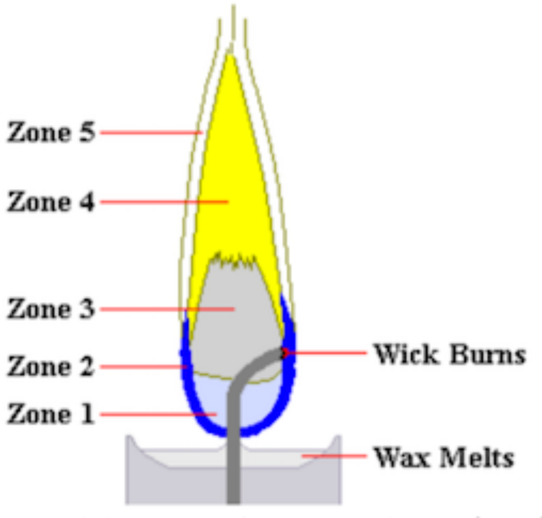


Image 3: Photo Credits: Richael E. Barrans

Modeling

For our project, we modeled our entire scene using Blender. The first step was to model the candle wax, wick, and candle flame. The candle wax was modeled as a cylinder with some deformations using the SculptDraw brush to create the melting wax look. The wick as modeled using a cylinder that deforms and follows a NurbsPath. Finally the candle flame was modeled as a stretched sphere, forming an ellipsoidal shape. The flame was subdivided three times with Catmull-Clark subdivision to create a smooth surface containing 3840 faces.

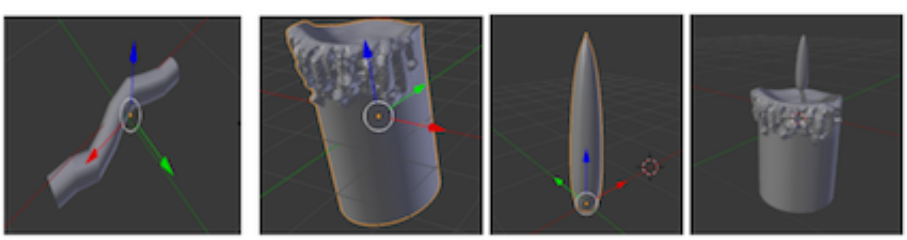


Image 4: Modeling of candle on Blender.

Next, we decided to model a candle stand to hold three candles to help increase the amount of illumination in the scene. The candle stand was created using many cylindrical scalings and extrusions. To create a more realistic look for the candle stand, many of the edges were beveled to smooth out sharp edges that made the candle stand look fake. The book was modeled using extruded bezier curves to make the pages. Finally, a small bowl was modeled using extruded circles.

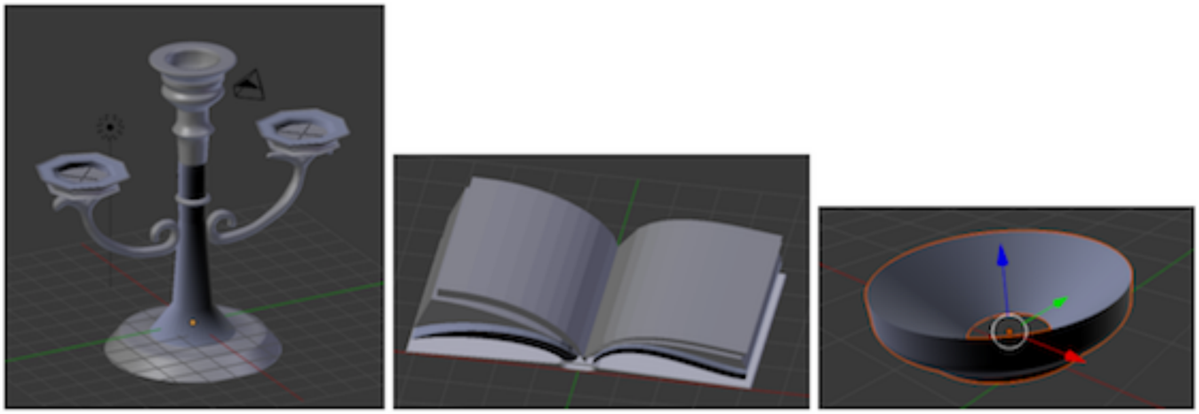
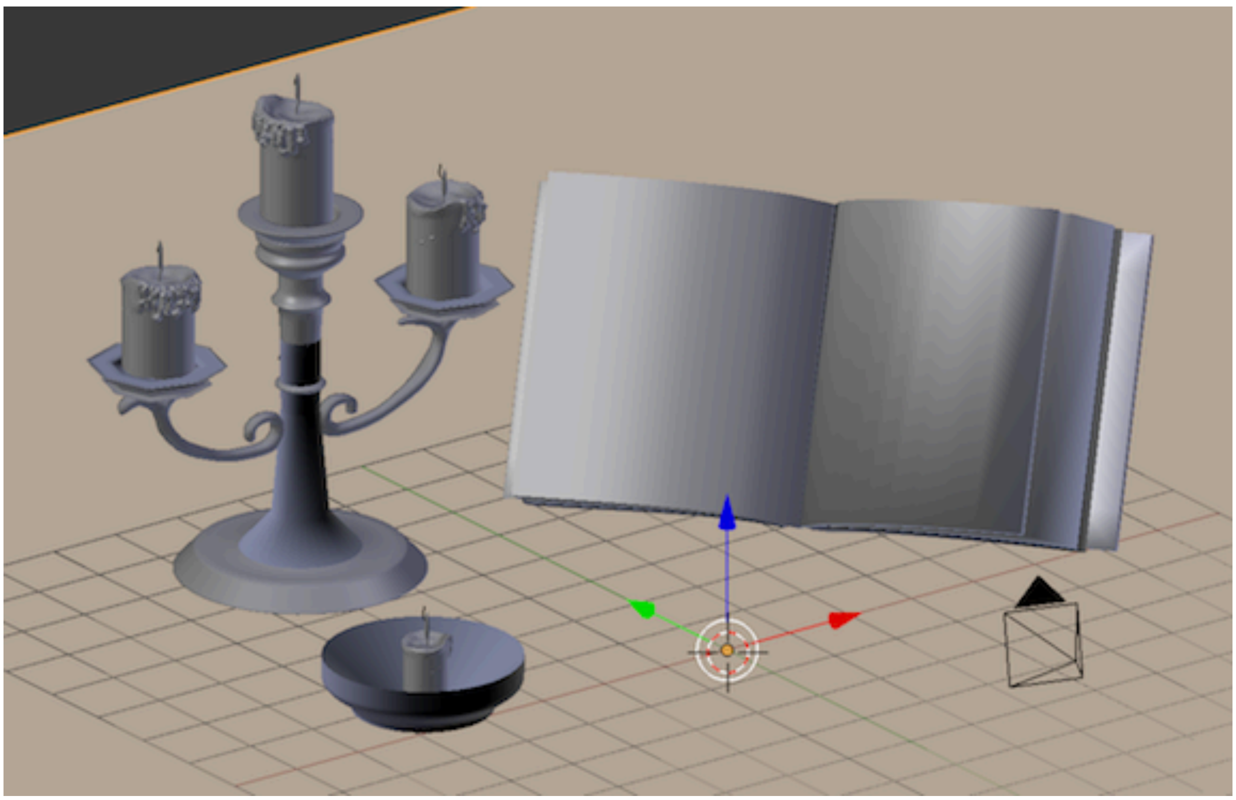


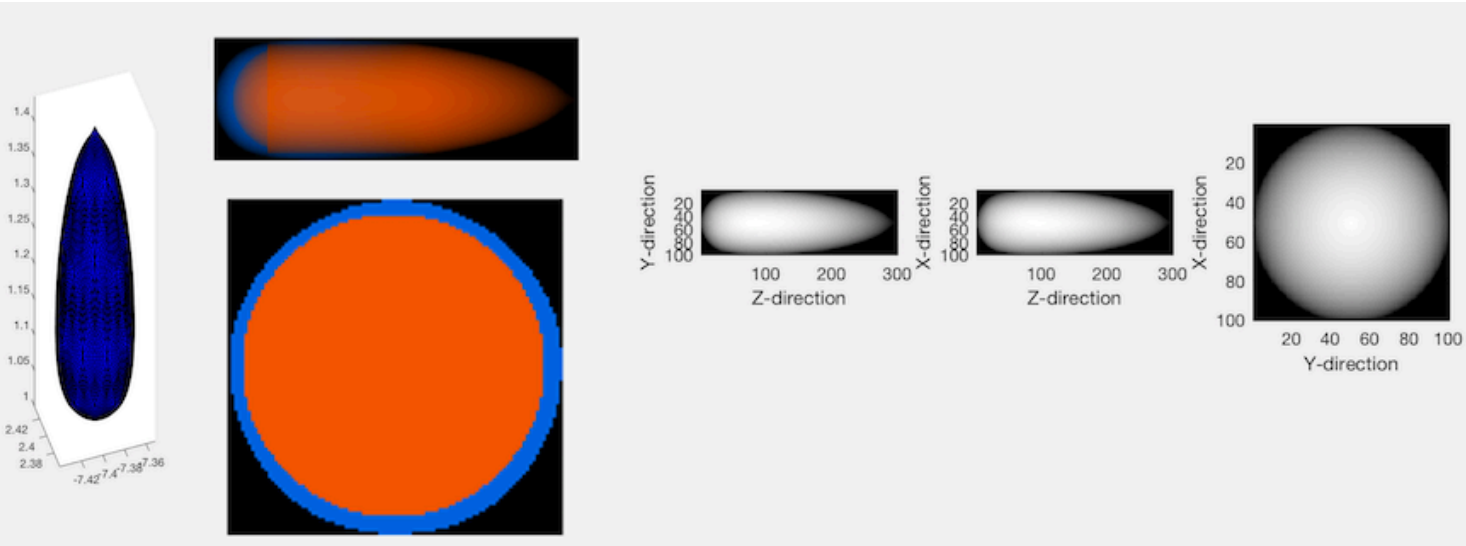
Image 5: Modeling candle stand, book, and bowl (left to right).

The final blender image is shown below.



Rendering

The first step behind rendering was to generate voxels for the fire. This was so the color and intensity of each position of the fire can be precomputed with MATLAB before importing it into PBRT to use to generate the fire. The voxelized version of the fire 3D model was done through the use of an open source MATLAB script. Once the fire model was voxelized, temperatures were manually added to the different regions of the candle to follow the temperature gradient of a real candlefire [3].



The main algorithm behind the color and intensity of the fire was the blackbody equation as follows:

$$B_\lambda(\lambda, T) = \frac{2hc^2}{\lambda^5} \frac{1}{e^{\frac{hc}{\lambda k_\mathrm{B} T}} - 1}$$

The equation was run for a given temperature for a vector of wavelengths that the human eye can see. From this we can approximate a Power Density Spectrum over these wavelengths. The algorithm defined in [4] was used to then generate an RGB value from the power spectrum.

The spectrum was first used to compute CIE numbers as follows:

$$\begin{aligned} X &= \Delta\lambda \sum_{\lambda=\lambda_{\mathrm{red}}}^{\lambda_{\mathrm{violet}}} \bar{x}_\lambda P(\lambda), \\ Y &= \Delta\lambda \sum_{\lambda=\lambda_{\mathrm{red}}}^{\lambda_{\mathrm{violet}}} \bar{y}_\lambda P(\lambda), \\ Z &= \Delta\lambda \sum_{\lambda=\lambda_{\mathrm{red}}}^{\lambda_{\mathrm{violet}}} \bar{z}_\lambda P(\lambda). \end{aligned}$$

The Y number was linearly scaled and range limited in order to get the intensity of the light. The CIE numbers were then used to compute approximate RGB values for the entire fire as follows:

$$\begin{bmatrix} x_r & x_g & x_b \\ y_r & y_g & y_b \\ z_r & z_g & z_b \end{bmatrix} \cdot \begin{bmatrix} J_r \\ J_g \\ J_b \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

The color and intensity of the light in the blue region as well as the dark region was set manually due to the fact that these regions are not primarily guided by blackbody equations. The bright regions was then designed to have a temperature of 2000K, which is the average candlelight fire temperature. After the intensity and RGB at that point was computed, a manual gradient was applied to the entire voxelization to provide smooth transitions and more a realistic fire. On PBRT, we had first created a subclass of the medium class using the GridDensityMedium class as reference. In a similar manner as the GridDensityMedium, voxels were read from a pbrt file and imported into float arrays. Two float arrays were created, one for the RGB values as a flattened 4D tensor, and one for the intensity values as a flattened 3D tensor. In addition to these, the medium class was modified to include virtual methods that returns the interpolated values of RGB and intensity at any given point in the voxel map, as well as a method to return the dimensions of the voxel map. The shape class was modified to include a couple methods that we used for our computations. A method returning the number of faces in a triangle mesh was included in the class, as well as a method for returning the position of the current face in a triangle mesh. The actual light source was created as a subclass of the area light source. In our light source, our data medium was used as an input variable, so that the light source can use the medium in its computations. The light source reads through the voxelization every time the L() function gets called. In order to get an accurate power spectrum given the position where the ray touches the shape, a line integral was computed from the ray position at the intersection through the entire fire shape as follows:

$$\int I(x, y, z)dl$$

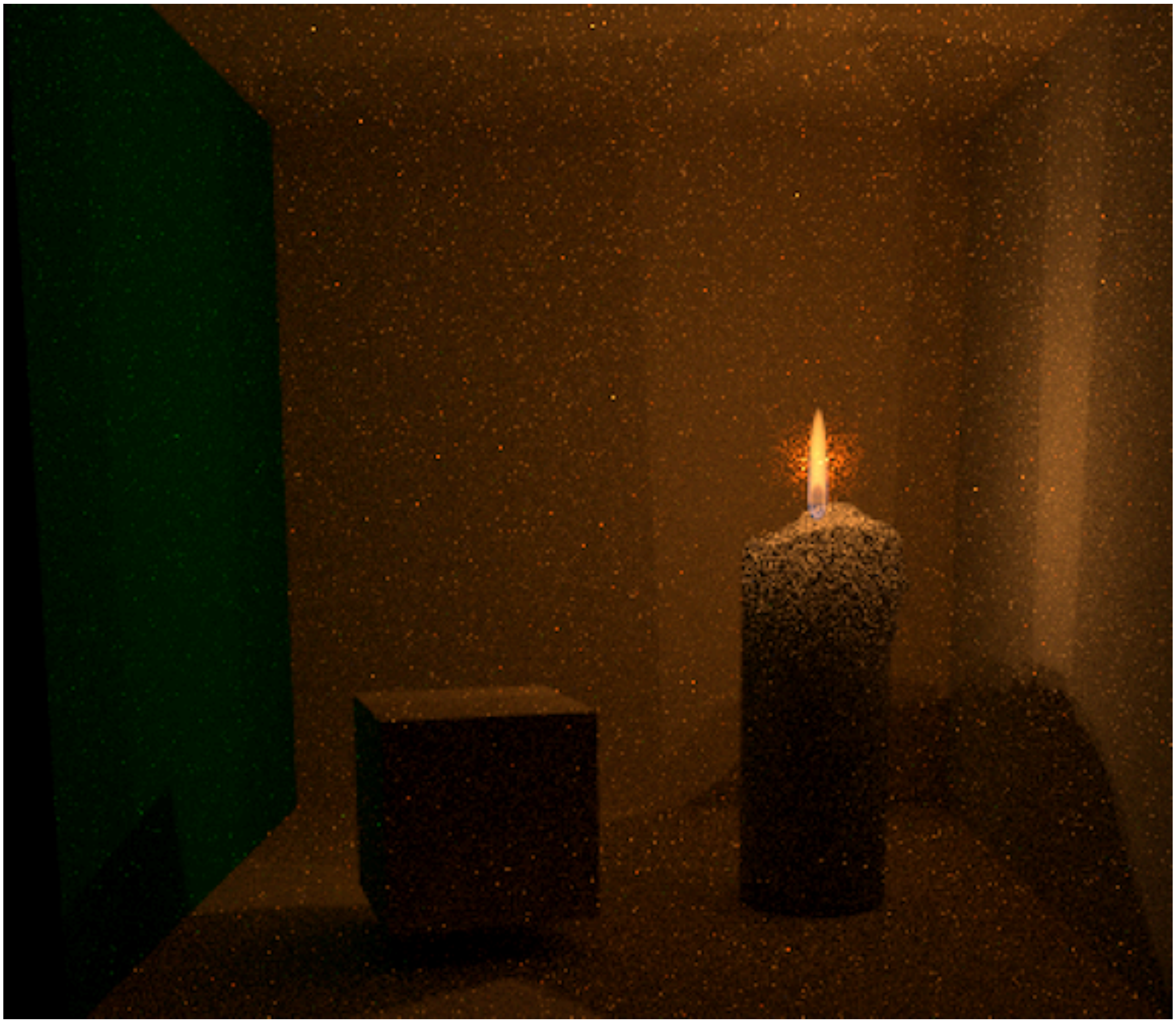
The line integral method we used was similar to the method used by Villemin and Hery in their BRDF sampling of a heterogenous medium [2]. This ensures that every voxel in the line contributes to the intensity at any given direction, so that all parts of the light source that shoots light in the same direction is taken into account. In order to ensure that the fire light does not oversaturate in color, it was kept very dim in brightness in order to keep the fire visible, but with 8 added point sources around the fire in order to provide ample light for the scene. The intensity and color of the point sources was generated through the use of sampling within the flame, and scaling the intensity by a constant.

After the light was generated, a glow medium was generated in order to add the glow effect around the lights. In essence our medium was a simple scattering medium that decreased in density linearly in distance from the fire. The voxel tensor was generated with Matlab, and included in the pbrt file as the exterior medium.

Key Challenges

A key challenge of this project was to generate a candle flame in which all layers of the fire are realistically represented, but at the same time provide ample lighting to the scene. Earlier iterations of the flame was able to provide good lighting by itself, but the fire itself looked completely white due to oversaturation. On the other hand, if the fire was set to be just dim enough to stop it from being oversaturated, it was unable to provide ample lighting to the scene. This problem was rectified by the use of point light sources alongside our fire light. The fire was set to be dim enough to show all the layers properly, and at the same time multiple point light sources were set around the fire to be the true source of illumination. 8 sources were used in order to mitigate strange behavior from shadows when less sources are used.

(Intensity too high, ran with large number of samples)

 (Not enough point light sources, noise was due to low samples)

Division of Labor

We began with Sang Goo looking up possible research papers to reference, while Viet began the modeling process on Blender. Sang Goo began working on the algorithms on MATLAB and PBRT and Viet joined him after the modeling process was complete. Both members alternated during the coding process and helped each other debug problems.

Results

For the final result, we performed halton sampling at 8192 pixel samples. This is important to fully create the glow on the candle flames and remove the noisy dots. We used the volpath integrator in order to make use of the subsurface scattering for the candle wax. Our image was rendered with a 1000x1000 resolution. The candle stand had a copper material, while a wood, paper, and clay texture was added to the table, book, and bowl respectively.

References

[1] Parker, Steven G., and Vincent Pegoraro. “Physically-Based Realistic Fire Rendering.” Scientific Computing and Imaging Institute, School of Computing, University of Utah.

[2] Hery, Christophe, and Ryusuke Villemin. “Multiple Importance Sampling for Emissive Effects.” Pixar Technical Memo 13-02.

[3] Barrans, Richard E., Jr., Ph.D. "Candle Flame." The Chemistry of Pysankarstvo. N.p., n.d. Web. 31 May 2017.

[4] Walker, John “Colour Rendering of Spectra” Web. 25 April 1996.