

Aurora Rendering with Sheet Modeling Technique

Yi Xiang Ng*

Abstract

Aurora, also known as the Northern Lights, is a form of fascinating display of Mother Nature. This luminous atmospheric phenomenon occurs most frequently above 60° North or South latitude, but also in other parts of Earth. In this project, based on the analysis and the algorithms stated in the paper [1] “Simulating the Aurora Borealis” by Baranoski et al, the purpose is to implement this rendering effect using OpenGL via simple sheet modeling technique that consists of sine wave coupled with Bezier curves for generating randomness. The result is in a simple and crude form of display of aurora polaris.

Keywords: Aurora, Bezier curve, sine curve

1 Introduction and motivation

Sun is the core energy source within the Solar System. Earth being part of this system, is incessantly showered by the rays emitted from the Sun. These rays are named as solar wind which consists of atomic particles, photons and electrons. Solar wind interacts with Earth’s magnetic field, especially near the Polar Regions. This interaction is in a form of collision with gaseous molecules in the Earth’s magnetosphere, thus resulting in excitation energy. Subsequently, this energy is lost via the emission of visible light during the night.

Being able to simulate this aurora is beneficial in terms of educational purposes or entertainment industry such as in game environment.

The purpose of this project is to examine the sheet modeling technique stated in [1] to generate simple and fast synthetic rendering of Aurora Polaris (lights) using C++ language with OpenGL API.

In the following sections, I will briefly elaborate on the scientific background of this lighting phenomenon. After which, the modeling approach used and its corresponding implementation. Compare with actual photos of actual Aurora Polaris to determine the shortcomings in the result phase. Finally, concluding with difficulties encountered during the project and viable enhancements in future implementation.

2 Review of Literature

2.1 Auroral Science

Understanding the background of this natural display is essential to generate realistic image of it.

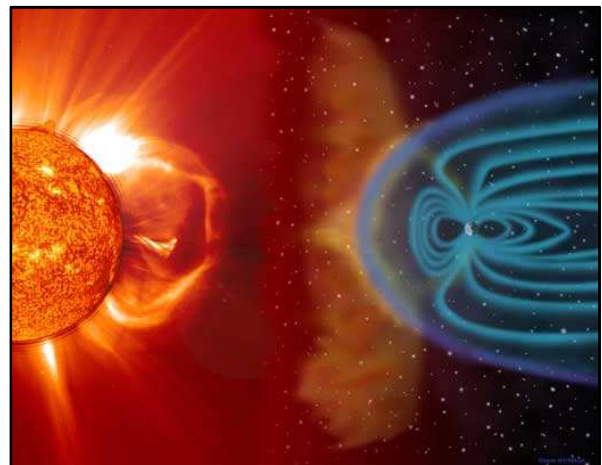


Figure 1: Interaction between Sun and Earth. Current flow in the magnetosphere during magnetic storm (Courtesy of NASA/Steele Hill)

*email ngy2@rpi.edu

The origin of this glamorous display is from the bursts of high-energy charged particles, mainly protons and electrons in the solar wind from the Sun. This incoming plasma causes magnetic storms to occur. They are signs of disturbance of our Earth's magnetic field. Figure 1 illustrates the main currents flow in blue color around the Earth during magnetic storms.

The energetic particles will precipitate towards the dense upper atmosphere of the Earth, near 100 to 300 km (63 to 190 mile) height in the ionosphere (Figure 2). Thus hitting and breaking apart oxygen and nitrogen molecules to form charged ions.

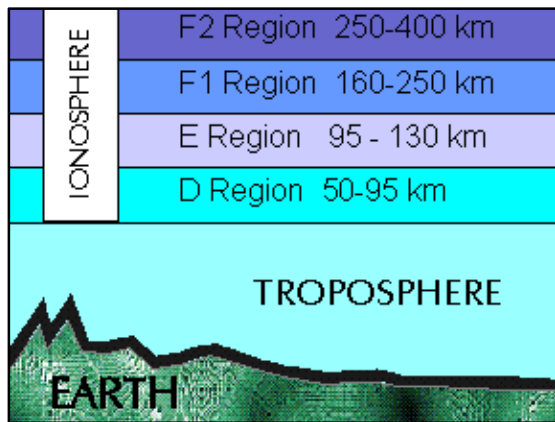


Figure 2: Ionosphere (Courtesy of www.atis.org/glossary/definition.aspx?id=4329)

The resulting ions release the excess energy from the collision as auroral lights with red, yellow, green, purple colors that are characteristics of the air molecules and the excitation energy. The collisions occur randomly. The most dominant aurora is the atomic oxygen "green line" with 557.7 nm in wavelength, around 100 km. Next, prominent color often spotted is red which is located in the upper parts caused by the atomic oxygen "red line" with 630.0 nm. In between these two lines, is the blue band from ionized nitrogen molecules with 427.8 nm.

The different colors in the auroral spectrum exhibit different behaviors. The transition state in the atomic green line only exists up till 0.7 seconds, whereas, the red line can exist up till 110 seconds. So atom can travel a much greater distance which resulted in a larger red emission area. However, more collisions also imply that the capacity of emitting light is also reduced as they are inversely proportional.

The shape of an aurora is influenced by the energy and density of the electrons entering Earth's atmosphere. The formation of falling electrons is often described as "curtain" of lights. It can bend and spiral along the composition forming rotational distortions. These curls are responsible in another auroral feature which is the electron beams/rays. Each ray is on average has a diameter of 1 km or less and run vertically down till several hundred kilometers. Varieties of auroral forms from ground observation are dependent on this ray as its fundamental building block.

The forms are classified into two groups. The first group consists of those without ray structure. The forms are arcs and bends or diffuse patches.

The second group consists of those with ray structure. The forms are rayed arc, rayed band, corona and drapery. The size of these forms usually extends up till 1000 km and width ranging from 10 km up till several kilometers. Vertically, they can be stretched between 20 km to 30 km.

2.2 Other resources

The foundation of the project is based on the prior work in C++ coding from the homework assignments in CSCI 4530 Advanced Computer Graphics at Rensselaer Polytechnic Institute (RPI), taught by Professor Barbara Cutler.

Additionally, VideoTutorialsRock.com has a series of very helpful tutorials with code snippets written in C++ and OpenGL. Specifically, one of which is stating how to perform texture mapping on a 2D surface.

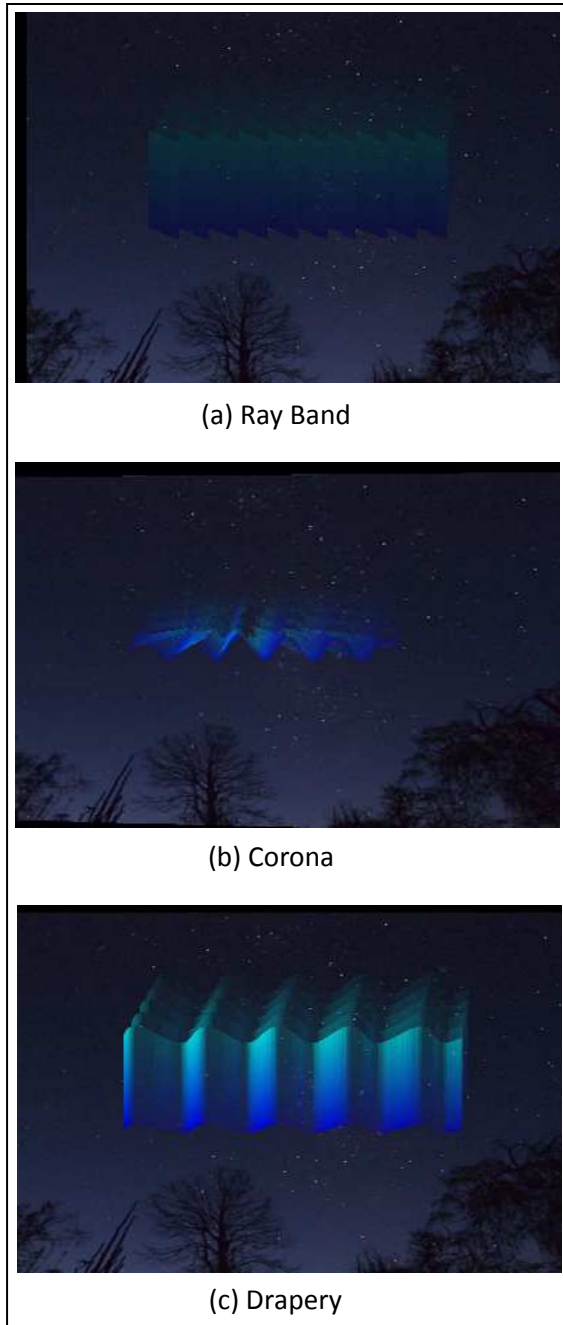


Figure 3: Images from project implementation for (a) ray band, (b) corona and (c) drapery. (Blue)

3 Sheet Modeling Technique

3.1 Aurora Representation

Essentially, the approach used in this project implementation is via sheet modeling technique. From previous auroral studies, the stream of falling electrons can be appropriately be represented by sheets and bounded by sine wave. The parameters for specifying the characteristics are the amplitude and phase of the sine wave.

With the base structure of the sheet formed, Bezier curves are used to replace at quasi-periodic interval to form the rotational distortions. This repetition refers to a form of motion that is regular but never exactly repeating. By varying the control points of the Bezier curves we can achieve both the bend and curl forms in the display.

After which, the precipitation of the rays is simply extruded downward according to the ratio stated in section 2.1. As for the thickness of the sheet, it is realized using accumulation buffer in OpenGL to simulate the blurring effect of the light. On top of the generation of sheets, suitable homogenous transformations such as translation, scaling and rotation are used to facilitate in the positioning of the auroral lighting.

Due to the time constraint, the night view in the scene is based on texture mapping on a 2D polygon for a straightforward illustration.

3.2 Aurora Implementation

In the actual implementation, the programming language used is C++ with OpenGL library utilities. The project used OpenGL quadrilaterals (GL_QUADS) to join the respective vertices together in the coordinate world.

Afterward, the translucent effect of the sine wave-based polygon formation is achieved by varying the alpha value, at ranging from 25% to 50% opacity only for the GL_QUADS.

Lastly, the blurring effect and width of the lighting are both rendered using the same scene multiple times but translating them with a small amount each time. With the help of the accumulation buffer in OpenGL, it can achieve the righting blending effect by having each sub-scene with an equal amount of weighting in the overall scene.

4 Difficulties and shortcomings

There are many flaws in this project. One of which is the lighting effect of the auroral lights. It was not implemented using appropriate methodologies such as volumetric photon mapping or ray tracing techniques.

Secondly, due to hardware limitation that I had, I was not able to implement any Shader methods to blend the color better or to utilize frame buffer object (FBO) in OpenGL to generate the glowing effect. Thirdly, Gaussian blur can also be added to the display to generate more realistic images.

Fourthly, I did not put into practice mathematical formulations stated in [1] for generating of the RGB color schema and the precipitation of the electron beams.

Lastly, the quasi-periodic folds are yet to be implemented in the current version of code. However, it will be added still after the submission deadline but before the presentation as much as possible.

5 Results

Although, these images are incomparable with actual auroras, they can still be utilized in

games or cartoonish environment for quick and fast rendering of such celestial lighting display. In this section, shown one more image generated from my project implementation.



Figure 4: Image from project implementation double curtains effect using Bezier curves (Green)

6 Conclusion

Although, the synthetic auroral images from this project implementation are not as realistic as it should be. Nevertheless, this piece of software implementation can be still utilize in simple animation or teaching aid to focus users' attention to certain portions of the display.

Future improvements stated in section 4 are definitely applicable to make this software implementation a much better emulation. So as to be more convincing as to overcome the faults stated previously. Usage of particle system in fluids can also help in simulating the random behavior of the falling electrons from the collisions.

Furthermore, this artificial auroral lighting can be merged with appropriate night rendering scene that has the right amount of night illumination. Thus generating a more comprehensive and appealing night scene.

Acknowledgement

I would like to thank Professor Barbara Cutler and Cody Philips (Teaching Assistant) for their valuable advises and insights along the way during this course (CSCI-4350 Advanced Computer Graphics) at RPI.

References

- [1] Baranoski, G.V.G., Rokne, J.G., Shirley, P., Trondsen, T. and Bastos, R. *Simulating the Aurora Borealis*. Computer Graphics and Applications, 2000
- [2] Baranoski, G.V.G, Wan, J., Rokne, J.G., Bell, I. *Simulating the Dynamics of Auroral Phenomena*. ACM Transactions on Graphics (TOG), 2005
- [3] Baranoski, G.V.G, Wan, J., Rokne, J.G., Bell, I. *Simulating the Dynamics of the Dancing Lights*. Technical Report CS-2002-16 April, 2002
- [4] Campbell, W. H. *Earth Magnetism: A Guided Tour through Magnetic Field*. Harcourt /Academics Press. 2001
- [5] Foster, N. and Metaxas, D. *Realistic Animation of Liquids*. Graphical Models and Image Processing, v.58 n.5, p.471-483. 1996
- [6] Henrik, W.J. & et al. *A Physically-Based Night Sky Model*. SIGGRAPH'2001.
- [7] Henrik, W.J. & et al. *Night Rendering*. Tech. Rep. UUCS-00-016, Computer Science Department, University of Utah, August 2000.
- [8] Jarosz, W., Zwicker, M., Jensen, H.W. *The Beam Radiance Estimate for Volumetric Photon Mapping*. Computer Graphics Forum, 2008
- [9] Petrie, W. *Keoeit – The Story of the Aurora Borealis*. Pergamon Press. 1963.