Illuminance plot with OpenGL

Cagin Agirdemir
Computer Animation and Game Technologies
Hacettepe University Information Institute
Ankara, Turkey
caginagirdemir@gmail.com

Abstract

This study is a simulation program that calculates the effect area of different light sources on a plane. In this project, it is possible for the user to visualize the positions of different light sources in 3D. User can change the angle of view through the help of mouse. Light contributions are calculated on the pixel is selected by the user. The amount of light from each source is shown on the console screen.

Keywords

3d, Open Source, illuminance simulation.

I. Introduction

The use of lighting simulation software is commonplace amongst professionals concerned with predicting the illuminance within buildings and for external environments. Although numerous programs exist, the mathematical algorithms used by most rely on either ray-tracing or radiosity methods to solve the global Illuminance equations. Dialux, Relux, AGI32 and Radiance are amongst the most poplarly used programs for lighting simulations.[1]

Each of the software programs has their photometric results compared for a series of scenarios. The scenarios include a daylight only scene, daylight and a reflected artificial light scene and a reflected artificial only light scene. [1]

Although many are commercial software, there are very few open source lighting simulation software. The most well-known of these is Radiance. It was developed in Lawrence Berkeley National Laboratory in 1985. Today it is available as a desktop version. It is published on https://radiance-online.org/ site.

Most lighting software includes similar features such as point-by-point calculations and daylight studies. All produce renderings, although at differing levels of quality and photo-realism. [2]

In this study, a program based on open source that can simulate single plane and multiple light sources will be describe. C ++ is used as development software language and OpenGL is used as graphic libraries.

II. OpenGL Settings

Firstly, the x and y axis lengths of the plane to be produced are requested from the user. These are followed by the opening of two screens. One of these screens is the 3D environment , the other screen is 2D. Second screen is represented plane, and it is 100 times more detailed version of normal plane

User can change camera direction through help of the mouse. User looks at this screen as frustum.

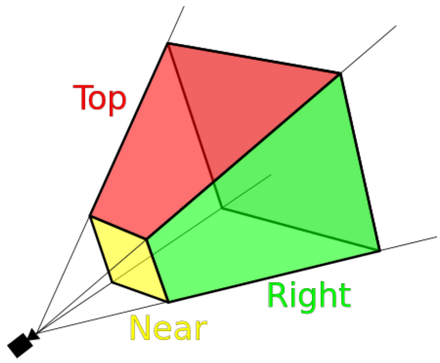


Figure 1 : Viewing Frustum

User will see a pattern on 2D screen window. This pattern contains two axes and 3 circles starting from the center. In this way, it helps the user aware where the pixel you choose is.

III. Implementation

User can add a new light source via right click menu. After pressing the light source add button, features related to the light source are requested on the console screen. These features are,

- Distance between surface are and light source
- Bulb type
  - Standard
  - Halogen
  - CFL
  - LED
- Bulb power as watt
- Position as x

- Position as y

A. *Representation of light source in 3D environment*

Wire sphere with a diameter of 1 cm is drawn in the coordinates specified by the user. Also, the x,y and z coordinates of the sphere are indicated by mini colored lines.

The user can add up to 4 light source. The effect that each light source has on its surroundings is shown circularly at the base. Visibility of the circles is decreased that is depend on power of light due to distance.

To better display the location of the light source within the screen, a line is drawn from the light source to the base and the middle of the plane.

B. *Determining the light value of a pixel on the plane*

After the light sources are placed, 2d screen is used to calculate the light effect on any pixel. The pixels clicked on the 2d screen are displayed on the console screen and lines are drawn from that light sources to that point.

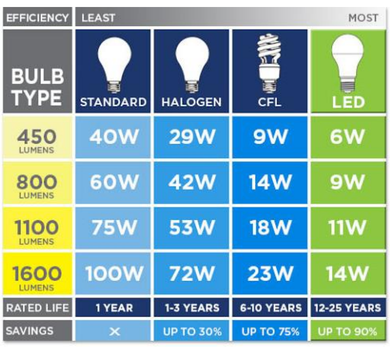
C. *Equations*

The calculation of illuminance at a certain point is simply calculated as follows.

$$E = \frac{I(\cos\alpha)^3}{h^2}$$

where α is angle of incident from the light source and *h* is the vertical distance from the light source to the plan of incident.

Power and bulb type data which is entered by the user are used to calculate. A table is shown as figure 2 used for this calculation. Because the bulb wattage values are limited to the products of the manufacturers and are generally produced at certain values. For this reason, the table below had been used while taking the lumen values. [2]



EFFICIENCY	LEAST			MOST
BULB TYPE	STANDARD	HALOGEN	CFL	LED
450 LUMENS	40W	29W	9W	6W
800 LUMENS	60W	42W	14W	9W
1100 LUMENS	75W	53W	18W	11W
1600 LUMENS	100W	72W	23W	14W
RATED LIFE	1 YEAR	1-3 YEARS	6-10 YEARS	12-25 YEARS
SAVINGS	×	UP TO 30%	UP TO 75%	UP TO 90%

Figure 2 : Lumen table [3]

Another formula used is calculating the angle between two vectors. 3d angle calculation is used and the following is shown.

$$\cos\theta = \frac{a\,.\,b}{|a|.\,|b|}$$

where a vector (red strip) is between light source and specified point; b vector (red) is between light source to based.

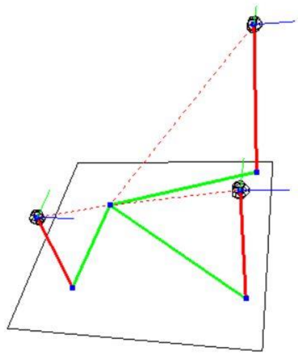


Figure 3 : A and B vectors in angle calculation

* colors has been inverted in terms of compatibility with paper background.

$$E = \sum \frac{I_i(\cos\alpha_i)^3}{h^2}$$

The above formula is used to calculate all light effects in a pixel.

D. *Light Standarts*

When prompted by the program, the user should indicate via console what kind of place user had using for plane. Depending on the user’s chose location, it will determine the minimum lux value from which it comes from the minimum lux value in the standards. There are many organizations and regulations that determine lux standards. Only Illumination Engineering Society (IES) standards were used in this study. Established in 1906, the Illumination Engineering Society is the recognized technical and educational authority on illuminance.[4] In this study, only Illuminance values are presented for certain places. These values and lighting values are shown in the table below.

Illumination Values (Lux) Table [5]		
Place	***Average***	***Range***
Lunch & Break Room	107.63	53.8 - 215.2
Classroom (Challenging Applications)	269.09	269 - 1076.3
Classroom (Typical Applications)	161.45	161.4 - 645.8
Open Office (Desk)	430.55	322.9 - 538.1

IV. Results

At this stage, the illuminance value calculation of the the pointly output was applied to the entire plane. A progressive color scale from red to yellow is were used to distinguish illuminate values from each other. The value of each color transition corresponds to 5-20 lux.

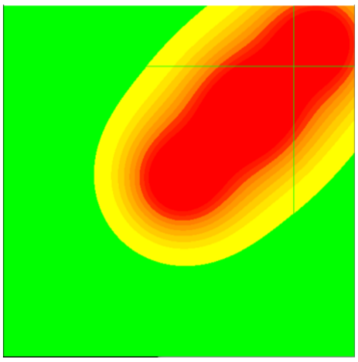


Figure 4 : Illumination plot through 3 source

In the output shown in Figure 4, shown the eligible space in terms of Illuminance standards of classroom (Challenging Applications).

The color transitions from red to yellow show the regions where the Illuminance value should be increased. This boost can be done by the power of the light source or the type of bulb. However, it will be more appropriate to use a new light source for green areas.

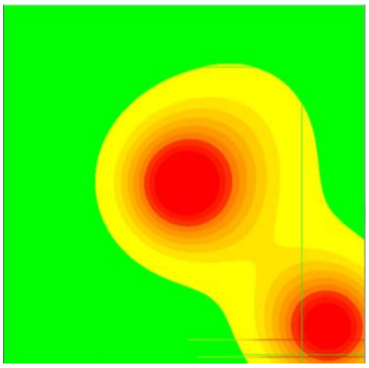


Figure 5 : Illumination plot through 2 source

In another example, the program output is figure 5. This examples shown the result of two slightly spaced light source. Also, some distortions appear in the lower left corner.

Along with the whole are Illuminance plot, the point illuminance value calculation function works. Thus, Illuminance contribution on the regions for every source can be seen in the console.

V. Conclusion

These and similar projects can generally be used within the scope of studies on light source. this may be a game lighting design or a physical one. An example of this is "Simulation and Comparison of the Lighting Efficiency for Household Illumination with LEDs and Fluorescent Lamps " Project. [6]

In this example, where LED and Fluorescent lamp effects are compared, it is seen that the lighting plots made on the surfaces are drawn.

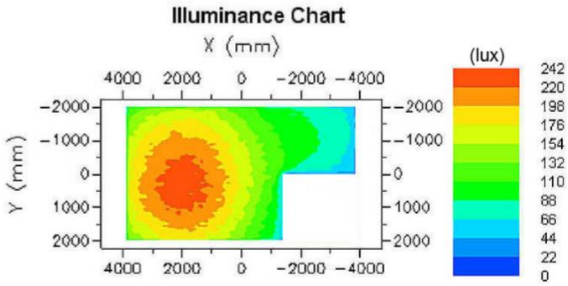


Figure 6 : Output of a commercial software [6]

Looking at the lighting output in Figure 6, it can be seen that it has similar output characteristics. In these commercial programs, the development features are closed against surface calculations and their engines are not added. However, this study remains at the introductory level and is open to development for more features.

But if we look at the general features of lighting programs. On the basis of all of them, it is seen that ray tracing and point-to-point calculation can be made. When this lighting program and commercial programs are compared, it is seen that many features are missing. However to calculation of illuminance at any point on plane, it takes is an important basic feature.

References

[1] Peter Byrne, “Comparison Study of Four Popular Lighting Simulation Software Programs” , pp. 1-2, Feburary 2015.

[2] https://www.archlighting.com/technology/lighting-software-tools_o July 2020.

[3] http://www.westinghouselighting.com/lighting-education/brightness-lumens.aspx July 2020.

[4] https://www.ies.org/about/ July 2020.

[5] https://www.ies.org/standards/ies-lighting-library/ July 2020.

[6] Wen-Shing Sun 1 , Chuen-Lin Tien 2 *, Jui-Wen Pan 3 , Tsung-Hsun Yang 1 , Chih-Hsuan Tsuei 1 , and Yi-Han Huang, “Simulation and Comparison of the Lighting Efficiency for Household Illumination with LEDs and Fluorescent Lamps” , pp 6, August 2013.

VI. UPDATES

This update objective demonstrate diffusion effect on newly added second plane.

I also add to second plane where 3d scene window. It located 90 degrees turned from main plane and 3 unit shifted on x axis and -3 unit shifted on z axis.

Second screen is divided to 2 part though right click menu. In the opening, screen set a point at first plane. As user click plane 2 on option at right click menu, window set as control plane 2. In this way, user add new point on plane 2.

$$I_{diff} = K_d l_i \vec{a}\vec{c}$$

$$E_{sum} = \frac{I_S \cos\frac{ab}{|a|.|b|}}{d^2} + \frac{I_{diff} \cos\frac{cd}{|c|.|d|}}{d^2}$$