Human Systems Engineering and Design (IHSED2025), Vol. 198, 2025, 119–125
https://doi.org/10.54941/ahfe1006768

Calibration of Illuminance in Virtual Engine Light Rendering Based on Lighting Simulation Values

Chen Cheng and Zhou Xiaozhou

School of Mechanical Engineering, Southeast University, Nanjing, Jiangsu 210096, China

ABSTRACT

With the continuous improvement of computers' ability to recreate real environments, when immersive virtual reality devices are used for lighting design, approximate models are often adopted due to real - time rendering performance limitations. This leads to designs relying on subjective perception, making it difficult to achieve precise quantitative verification. Professional lighting design software can perform physically accurate lighting calculations, and virtual engines can conduct efficient real - time rendering. Combining the advantages of the two makes it possible to achieve correct calculation and visual verification of light distribution in the engine. This paper proposes a new method of using Unity HDRP as a lighting design tool. Through in - depth research on the rendering pipeline and setting key parameters, correct light distribution and calculation can be carried out in the game engine. By creating the same virtual environment, the brightness distributions on the same surface calculated by Unity and Dialux are compared. The results show that after external parameter calibration, Unity is reliable in reproducing light distribution, and combined with real - time rendering, it can conduct correct and visual lighting design.

Keywords: Lighting calculation, Real - time rendering, Lighting design, Dialux, Unity

INTRODUCTION

Lighting design, as a core element in fields such as architectural environments and industrial design, directly influences the functionality, aesthetic appeal, and user experience of a space. The traditional lighting design process often relies on professional lighting software for pre - calculation and static preview, with a focus on how to reconstruct real - world light (Heydarian et al., 2015). This working mode has obvious limitations: first, the cost of design modification is high, as each adjustment requires recalculation and remodeling; second, the static preview cannot fully present the dynamic light - shadow effects and the actual impact of human activities on the lighting environment; finally, designers are often limited by the expressive ability of 2D drawings and static renderings when communicating with users.

With the exponential improvement in the computing power of graphics processing units (GPUs) and the popularization of real - time ray - tracing technology, game engine technology is breaking through the traditional boundaries of entertainment applications and expanding into professional fields such as architectural visualization (Dunston et al., 2011), industrial design, and film and television production (Scorpio et al., 2022).

Mainstream game engines such as Unity and Unreal Engine, by introducing physically - based rendering (PBR) pipelines, high - dynamic - range imaging (HDRP), and real - time global illumination technology, have been able to achieve lighting effects close to photo - realistic quality while maintaining real - time interactivity. A new method called BIM - based Lighting Design Feedback (BLDF) was proposed (Natephra et al., 2017). This method uses Unreal Engine 4 as an interactive and immersive virtual environment, and utilizes a user - friendly interface to visualize lighting conditions and calculate lighting energy consumption. Users can set different parameters and perform simulations until a satisfactory design result is achieved.

Existing research mainly focuses on the optimization of game engines in terms of visual effects and artistic presentation (Burley et al., 2012), while relatively few studies have been conducted on the verification of quantitative accuracy in the application of lighting engineering.

A new method for measuring daylighting (Chamilothori et al., 2019) involves adding physically - based rendering effects in an immersive virtual environment. By using Radiance to render HDR images, the light distribution inside the considered room is obtained.

In addition, how to establish an accurate mapping relationship between the lighting parameters in the game engine and real physical quantities, and how to verify the reliability of its calculation results, remain key technical issues in lighting design using game engines.

One of the most prominent issues in the aforementioned research is to generate a virtual environment that is as close as possible to the real - world scene. Narrowing the gap between the virtual reality environment and the real environment is to provide users with a high level of “presence” (Schubert et al., 2003), “realism” (Maffei et al., 2016), and the “sense of immersion” of being “in - situation” in the virtual world, as if being in a real scene (Slater et al., 2009).

Therefore, this study aims to establish a set of parameter configuration methods and calibration procedures suitable for lighting design by deeply analyzing the light calculation mechanism of the Unity HDRP rendering pipeline. And through comparative verification with professional lighting software, the corresponding relationship between physical quantities and rendering parameters will ultimately be established.

Implementation of Artificial Lighting and Illuminance Calculation in Unity Real - Time Rendering Engine

Image rendering usually uses empirical - based lighting models. When conducting lighting design, it is impossible to quantitatively describe the scene illumination. Therefore, it is necessary to start from the principles, extract the illumination calculation process of the real - time rendering engine, and explore its capability in reproducing the illuminance distribution of lights in the real - world environment.

The proposed analysis method aims to determine the key parameters that can achieve light calibration, evaluate the impact of these parameters on the simulation of light distribution, and propose the optimal solutions for settings in Unity HDRP, so as to ensure that the simulation of artificial light distribution from the optical perspective has appropriate accuracy.

In Unity, the rendering pipeline processes the illumination information and geometric information in the scene, and the calculation of illumination follows physical formulas:

$$E = \Phi / (A \cdot cos\theta) \quad (1)$$

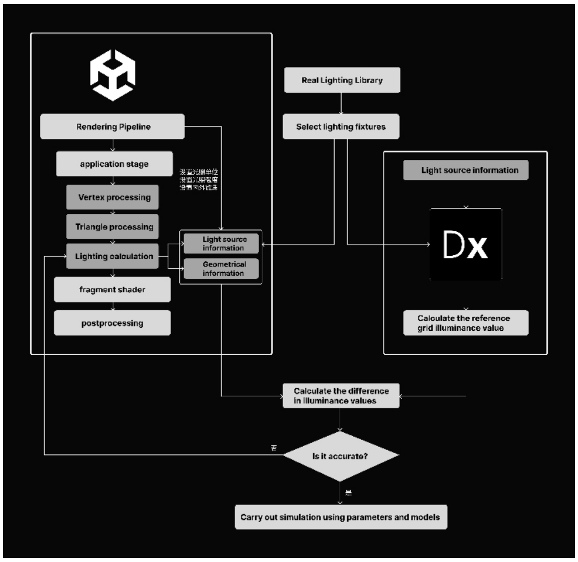

Figure 1: Technical roadmap for illuminance calibration of real-time rendering engine lights.

Through research on the rendering pipeline, information related to illuminance calculation can be extracted, providing data for subsequent studies.

Mapping Link From Illuminance Calculation to Screen Display

To implement illuminance calculation in the engine, it is necessary to understand the relationship between surface illuminance and the final displayed pixel brightness, the process of optical illuminance calculation, and the transmission process of lighting data in the engine.

The pixels ultimately presented on the screen by display devices go through multiple stages, a process involving multiple levels of physical optics, computer graphics, and display technology. The main links can be summarized as follows:

Stage 1: Light emitted by the light source (lm) → Scene illuminance (lux) → Object reflectivity → Scene luminance (cd/m^2)

Stage 2: Scene luminance (cd/m^2) → HDR value (engine processing) → Tone Mapping → LDR value

Stage 3: LDR value → Gamma correction → Display brightness value (0-255) → Screen physical brightness (nits)

The proposed calibration method will focus on the calculations related to illuminance before tone mapping, i.e., the first stage.

Description of Virtual Light Sources in the Rendering Pipeline

Unity's rendering pipeline classifies and describes light sources, controlling them with different parameters, which to a certain extent simulates the forms of real-world lamps.

(1) Types of light sources (analysis from the perspectives of code implementation and real lamp performance)

According to the panel displayed by Unity, the light sources in the engine can be mainly divided into point lights, spotlights, area lights, and ambient lights.

A "point light" is approximated as an infinitesimally small point emitting light in all directions, and its corresponding surface illuminance is calculated as: illuminance = luminous intensity (cd) / distance^2 $\times$ cos(θ);

A "spotlight" has a certain directionality and, depending on different IES files, exhibits different attenuation characteristics. Its corresponding surface illuminance is calculated as: illuminance = lamp luminous flux/ (2π $\times$ (1 - cos(spotAngle/2)));

An "area light" emits light from a rectangle, similar to a spotlight but with a cone angle set to 180°. Its calculation is more complex and will not be elaborated here.

This paper only discusses the performance of real-time lighting and does not consider techniques such as light baking and lightmaps. Based on the performance characteristics of real-world lamps, most lamps emit light in ordinary spaces with directionality, and the corresponding spotlight is the most suitable object for this research.

The engine supports loading IES files, which allows adjustment of the inner and outer cone angles, control over the attenuation range of the overall lighting effect, and customization of the lighting calculation in the rendering pipeline by modifying the engine's lighting calculation code.

HDRP supports the use of IES files to control lights. The engine parses light attenuation into a texture for graphic masking, enabling the reproduction of the image characteristic distribution of the light.

The inner and outer cone angles have a significant impact on the light. When no IES file is used, the inner and outer cone angles fully control the light distribution: the inner cone area is a fully illuminated region, and the area between the outer and inner cones is a gradient region.

When an IES file is used, the inner and outer cone angles act as constraints for the IES distribution, within which the IES data is "mapped". The outer cone angle becomes the clipping boundary of the IES distribution, while the inner cone angle has little impact on the IES distribution—it only affects the mixing of the central region of the IES data without altering the actual shape of the light distribution curve. The effect of the inner and outer cone angles on spotlighting is shown in the figure.

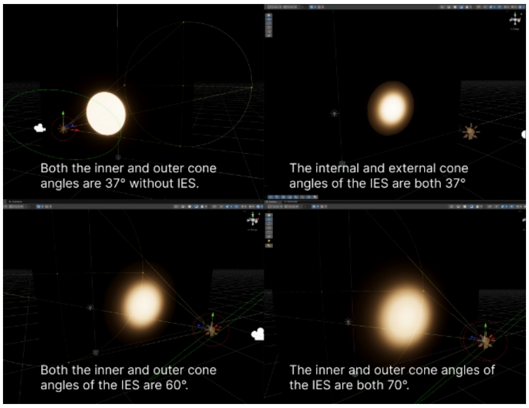

Figure 2: Application of IES and role of inner and outer cone angles in unity's spotlight.

Adjusting Unity HDRP to Replicate Light Illuminance Distribution With Reference to Dialux Illuminance Calculations

There is a certain mapping relationship between the graphics rendering engine when executing the final rasterized image and the actual physical quantities. The approximate data of the real-time rendering engine is calibrated with external accurate data, enabling it to correctly display the light distribution under the performance of real-time rendering (Ciampi et al., 2015).

To understand the engine's capability to physically reproduce light distribution, the following information needs to be obtained:

(1) Light source modeling method

Light sources in Unity exist in the form of structures, which contain information such as the spatial position, direction, luminous intensity (lm), and color temperature of the light. For illuminance calculation, the main information used includes position, luminous intensity, and the geometric information of the illuminated surface.

(2) Configurable parameters in the program

Unity provides some data interfaces, allowing the setting of light intensity through physical quantities such as candela (cd) and lumen (lm), control of light attenuation through inner and outer cone angles and range, and creation of surfaces to obtain position information of surface points at certain distances, which provides data for subsequent lighting calculations. The engine calibrated from an optical perspective can largely maintain the accuracy and visual performance in lighting design.

Dialux is a professional lighting design software (Scorpio et al., 2022), which has good accuracy in simulating point light sources, area light sources, diffuse reflection, and multiple reflections. Based on this, with Dialux as the benchmark, the illuminance information of the reference grid surface in the virtual environment extracted from Dialux is used as calibration data to evaluate Unity's capability in reproducing light distribution. By importing IES files into the scene and setting appropriate calculation elements (usually a plane), the illuminance information on the calculation element can be obtained. The number of sampling points for the calculation element can be set, and here it is set to 20 × 20.

The same virtual environment, a 2 × 2.4 × 3m darkroom, is constructed in both Dialux and Unity. A 100mm square grid is made on the target wall to calculate the luminance value, and the surface is highly diffuse reflective.

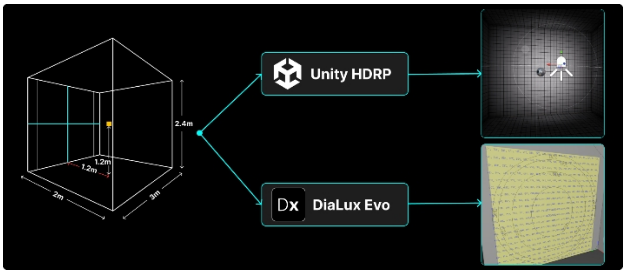

Figure 3: Construct the same virtual environment in unity and dialux.

Configure the lights in the virtual environments of Unity and Dialux using the same IES file, set up the same sampling grid to collect illuminance information, and draw attenuation curves based on the illuminance information. Calculate the differences between the obtained illuminance data values.

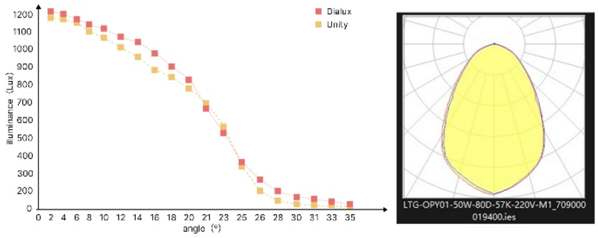

Figure 4: Illuminance attenuation curves of unity and dialux.

From the attenuation curves plotted based on illuminance, it can be observed that the illuminance values in Unity are slightly lower than those in Dialux. When reaching 21 degrees, the values in Unity and Dialux are the closest. In subsequent steps, more data will be used to adjust the values in Unity, enabling Unity to achieve the illuminance calculation accuracy of Dialux while meeting the requirements of efficient and high-quality real-time rendering.

CONCLUSION

Current real-time rendering engines can largely reproduce the visual performance of real-world lighting, but there are some errors in numerical values. The illuminance calculation errors can be reduced by measuring and adjusting the calculated values in Unity. After calibrating Unity with Dialux, it can correctly calculate the distribution of artificial lighting while ensuring the efficiency of real-time rendering, and provide reference values for lighting design.

REFERENCES

Burley, B. and Studios, W. D. A., 2012, August. Physically-based shading at disney. In Acm siggraph (Vol. 2012, No. 2012, pp. 1–7). vol. 2012.
Chamilothori, K., Wienold, J. and Andersen, M., 2019. Adequacy of immersive virtual reality for the perception of daylit spaces: Comparison of real and virtual environments. LEUKOS, 15 (2–3), 203–226 [online]
Ciampi, G., Rosato, A., Scorpio, M. and Sibilio, S., 2015. Retrofitting solutions for energy saving in a historical building lighting system. Energy Procedia, 78, pp. 2669–2674.
Dunston, P. S., Arns, L. L., Mcglothlin, J. D., Lasker, G. C. and Kushner, A. G., 2011. An immersive virtual reality mock-up for design review of hospital patient rooms. In Collaborative design in virtual environments (pp. 167–176). Dordrecht: Springer Netherlands.
Heydarian, A., Carneiro, J. P., Gerber, D. and Becerik-Gerber, B., 2015. Immersive virtual environments, understanding the impact of design features and occupant choice upon lighting for building performance. Building and Environment, 89, pp. 217–228.
Maffei, L., Masullo, M., Pascale, A., Ruggiero, G. and Romero, V. P., 2016. Immersive virtual reality in community planning: Acoustic and visual congruence of simulated vs real world. Sustainable Cities and Society, 27, pp. 338–345.
Natephra, W., Motamedi, A., Fukuda, T. and Yabuki, N., 2017. Integrating building information modeling and virtual reality development engines for building indoor lighting design. Visualization in Engineering, 5(1), p. 19.
Scorpio, M., Laffi, R., Teimoorzadeh, A., Ciampi, G., Masullo, M. and Sibilio, S., 2022. A calibration methodology for light sources aimed at using immersive virtual reality game engine as a tool for lighting design in buildings. Journal of Building Engineering, 48, p. 103998.
Schubert, T. W., 2003. The sense of presence in virtual environments: A three-component scale measuring spatial presence, involvement, and realness. Z. für Medienpsychologie, 15(2), pp. 69–71.
Slater, M., 2009. Place illusion and plausibility can lead to realistic behaviour in immersive virtual environments. Philosophical Transactions of the Royal Society B: Biological Sciences, 364(1535), pp. 3549–3557.