Area Light Sources in Cyberpunk 2077

Peter Sikachev
peter.sikachev@gmail.com
Independent
Warsaw, Poland

Giovanni De Francesco
giovanni.francesco@cdprojektred.com
CD PROJEKT RED
Warsaw, Poland

Kamil Nowakowski
kamil.nowakowski@cdprojektred.com
CD PROJEKT RED
Warsaw, Poland

Karol Kowalczyk
karol.kowalczyk@cdprojektred.com
CD PROJEKT RED
Warsaw, Poland

ABSTRACT

This talk discusses in detail the area lights technique used in Cyberpunk 2077. We outline the limitations and challenges we had, and then present our solution. Particularly, we describe our 100% analytical capsule lights, our *spot-capsule* solution, and our capsule light shadow technique. We then discuss our artistic pipeline, our performance results, and limitations.

CCS CONCEPTS

• **Computing methodologies → Reflectance modeling**; *Rasterization*; • **Applied computing → Computer games**.

KEYWORDS

area light, real-time rendering, capsule light, spot light, game development

ACM Reference Format:
Peter Sikachev, Giovanni De Francesco, Kamil Nowakowski, and Karol Kowalczyk. 2021. Area Light Sources in Cyberpunk 2077. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Talks (SIGGRAPH '21 Talks), August 09-13, 2021.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3450623.3464630

Figure 1: El Coyote Cojo, one of the iconic Cyberpunk 2077 locations. Notice the ubiquity of area lights in the environment.

1 INTRODUCTION

Cyberpunk 2077 is set in a futuristic world with a lot of artificial lighting. A lot of those lights are not fairly represented with point lights: fluorescent lamps in offices, emissive street ads, neon business signs to name a few. What makes things more difficult is that *most* of those lights are some sort of area light, meaning we need to be able to render a lot of them in the frame. Figure 1 shows a typical in-game location, full of area lights.

There are two state-of-the-art approaches to area lights: sphere and capsule lights in [Karis 2013] and rectangular lights in [Heitz et al. 2016]. Rectangular lights nicely fit a lot of real-world examples, like billboard ads and screens, but they have a few limitations which prevented us from using them. Firstly, they start to exhibit unstable behavior at grazing angles. Secondly, and most importantly, they are really expensive performance-wise: there is a lot of branching in the code that clips polygon against horizon and inverse trigonometry in the integration code.

Therefore, we based our solution on the approach outlined in [Karis 2013]. Below we discuss the improvements we needed to add to make capsule lights in an open-world game like Cyberpunk 2077 affordable on consoles.

2 IMPLEMENTATION AND RESULTS

Analytical Capsule Light [Karis 2013] focuses mainly on sphere light implementation. We quickly found out that most of our practical area light cases need capsule lights, as they are elongated along one of the axes.

However, the approach to capsule light implementation in [Karis 2013] did not directly work for us. Instead of an analytical solution for capsule lights, they effectively perform importance sampling with a low sample number. We could not do the same for two reasons. Firstly, this creates visible banding for very long light sources positioned parallel to the flat surfaces. Secondly, we could not afford evaluating BRDF more than once per light source per pixel.

We addressed these issues by introducing 100% analytical lights. The key idea behind this was to find the best representative incoming light direction and approximate energy modification based on the shape and orientation of the light. The main challenge here was

SIGGRAPH '21 Talks, August 09-13, 2021, Virtual Event, USA

ACM ISBN 978-1-4503-8373-8/21/08.
https://doi.org/10.1145/3450623.3464630

to come up with an energy term approximation that would create plausible results for most of the roughness-metalness combinations.

We pick the best representative direction as the closest point on the capsule to the ray between normal and reflected vectors, weighted by roughness. Next, we perform several adjustments: we attenuate based on the solid angle of the capsule compared to the solid angle of the sphere for less rough surfaces, then perform additional attenuation for grazing angles. Afterwards, we carry out a similar attenuation for the sphere radius. Figure 2 shows the results that all these approximations can achieve.

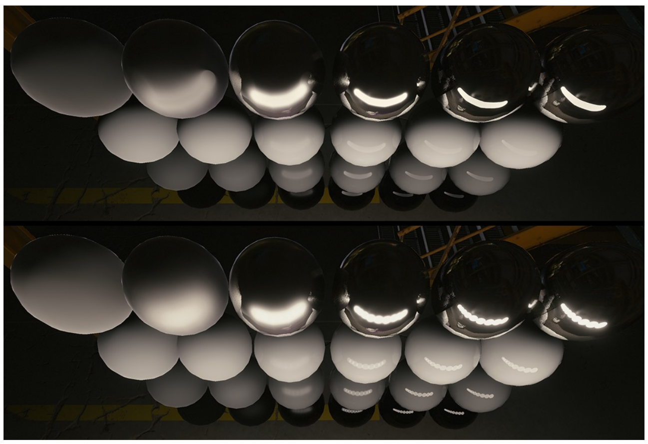

Figure 2: Comparison of analytical capsule light against a synthetic test (an array of sphere lights). Notice how energy is well preserved for a wide range of materials.

Spot Capsule Lights One of the advantages of [Heitz et al. 2016] is being able to create single-sided lights, which makes sense for objects like screens or glowing ads. We decided to add a similar functionality to capsule lights by introducing *spot capsule lights*. In a nutshell, we modulate light intensity at each point by using a spot light attenuation, with the virtual apex of the spot located at the closest point on the capsule axis and oriented orthogonal to it. This allows for multiple enhancements: the light can now be made of any arbitrary angle, light leaks can be more easily avoided, and fewer pixels need to be shaded thanks to tiled light culling. Figure 3 shows how the spot capsule concept works in practice.

Area Light Shadows While [Heitz et al. 2018] is an ultimate area light shadow solution, we needed an approach that would work for many lights per frame on less powerful hardware. We proposed to use perspective shadow maps, just for the spotlights, but with a few changes. We offset their position behind the light source so their field of view completely matched the field of view of our spot capsule light, and moved the front clipping plane to the light plane.

Area Light Performance and Optimization Using capsule lights instead of point lights yielded an approximate 30% performance overhead. However, adding the code to the lighting pass also increased the register pressure. To alleviate this, we factored out contact shadow logic to the shadow mask pass, which brought the vector general purpose register (VGPR) count back to the original value. For a standard material, the corresponding deferred lighting shader uses 64 VGPRs, whereas the most complex material utilizes up to 84 VGPRs. The deferred lighting pass for the scene from Figure 1 with around 60 area lights took about 2ms on Xbox One X at 2488×1400 light buffer resolution.

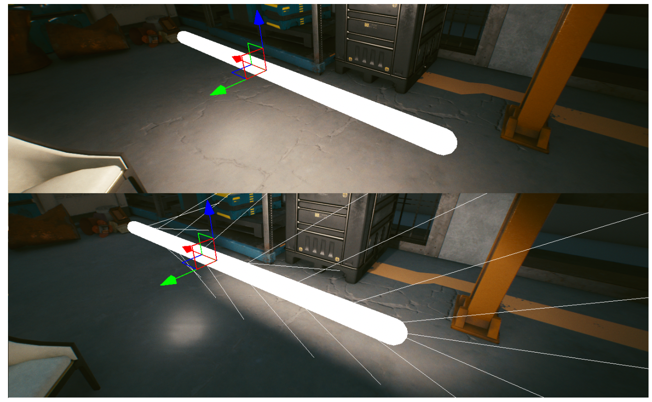

Figure 3: Regular capsule light (top) and spot capsule light (bottom). UI lines show the outer cone.

Artistic Implications Thanks to this particular set of features, our area light solution brought a great degree of artistic freedom in terms of usage when lighting the diverse environments in the game. The performance-friendly nature allowed us to utilize area lights in very high numbers in our locations without being much concerned about overloading the lighting pass. We were also able to use them in various open-world scenarios, such as street lamps, for example, where they become necessary to convey the cyberpunk aesthetics of the city.

The *spot capsule* feature gave us a great level of control, providing us with the ability to narrow and focus lighting emissions. In the first instance, we could avoid all issues related to light leaking, since we could not afford to have a high number of shadow-casting lights. Such an issue is very relevant in Cyberpunk 2077 due to the realistic design of spaces where each room or floor is a playable area. Secondly, it gave us a high level of precision when it came to painting surfaces with light, allowing greater artistic control. Together with radius cutoff, we were able to create "tunnels of light" which did not affect the environment, but were used solely to light characters.

Area lights used in conjunction with our portal light system, which gave local lights an ability to sample sky energy and color, was a major feature in allowing us to inject light coming from the sky in different areas of the open-world environment. Because of that, long alleys with low sky visibility were able to receive enough lighting from the sky utilizing just one large or long capsule light. The same trick was used in indoor spaces where we could convey skylight through large windows or doors.

REFERENCES

Eric Heitz, Jonathan Dupuy, Stephen Hill, and David Neubelt. 2016. Real-time polygonal-light shading with linearly transformed cosines. In *ACM Transactions on Graphics*.

Eric Heitz, Stephen Hill, and Morgan McGuire. 2018. Combining Analytic Direct Illumination and Stochastic Shadows. In *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*.

Brian Karis. 2013. Real Shading in Unreal Engine 4. In *SIGGRAPH 2013 Course: Physically Based Shading in Theory and Practice*. Anaheim, CA, USA.